SEGA

SEGA OF AMERICA, INC.
Consumer Products Division

Sound Development Manual

Doc. #ST-81-R2-031094

NOTICE

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.

2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without SEGA's permission.

3. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

4. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

5. No license is granted by implication or otherwise under any patents or other rights of any third party or SEGA Enterprises, Ltd.

6. This document is confidential. By accepting this document you acknowledge that you are bounded by the terms set forth in the non-disclosure and confidentiality agreement signed separately and/in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

(4/20/94 - 001)

READER CORRECTION/COMMENT SHEET

Keep us updated!

If you should come across any incorrect or outdated information while reading through the attached document, or come up with any questions or comments, please let us know so that we can make the required changes in subsequent revisions. Simply fill out all information below and return this form to the Developer Technical Support Manager at the address below. Please make more copies of this form if more space is needed. Thank you.

General Information:

Your Name ______________________ **Phone** ____________

Document number ST-81-R2-031094 **Date** ____________

Document name Sound Development Manual

Corrections:

Chpt.	pg. #	Correction

Questions/comments: ______________________

Where to send your corrections:

Fax: (415) 802-3963
Attn: Manager,
Developer Technical Support

Mail: SEGA OF AMERICA
Attn: Manager,
Developer Technical Support
275 Shoreline Dr. Ste 500
Redwood City, CA 94065

Section 1: General Information

1.0 System Outline

1.1 Hardware Requirements

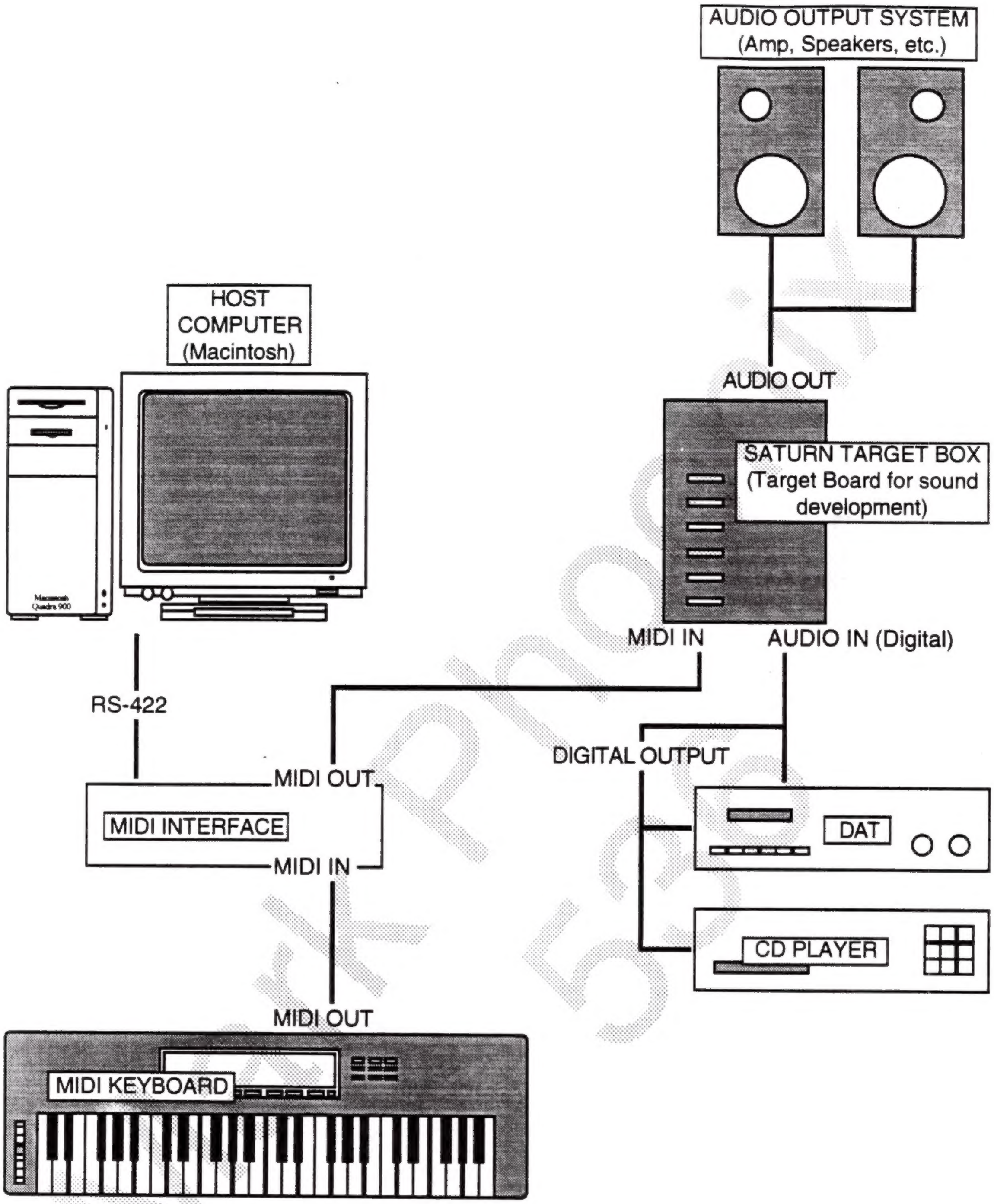
Device	Model	Description
Development host machine	Macintosh	Macintosh II series or later with SCSI interface. Operating system: KanjiTalk 7 or System 7 or later RAM: 16 MB or more HDD: 300 MB or larger and 1 GB when making HD recordings
Saturn Target box		Sound board is capable of operating alone.
MIDI instrument	MIDI keyboard, etc.	Instrument with MIDI output capability Used for tone development, composition, and creating sound effects
Audio equipment	CD player, DAT, etc.	Device capable of digital output Used for editing waveforms and HD recording
MIDI interface	Studio 5, etc.	MIDI interface for Macintosh

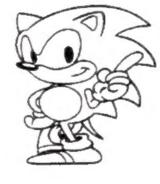
1.2 Software Requirements

System	Software	Source	Description
Tone development tool	Waveform editor Tone editor DSP linker	SEGA Market item SEGA SEGA	Used to edit waveforms and HD recordings Waveform edit tools available on the market can also be used (devices that support AIFF format such as Alchmy, SDII) FM/PCM tone development DSP program development
Composition tools	MIDI sequencer	Market item	Digital Performer (Mark of the Unicorn) Studio Vision (Opcode Systems) Cubase Audio (Steinberg), etc.
Sound development support system	Master system	SEGA	Map Tool

The tools listed below are for writing and changing sound drivers and other programs, and are not required for the development of sound data (tunes and sound effects).

System	Software	Source	Description
Program development tools	Text editor Assembler Linker Debugger	SEGA Market item SEGA SEGA SEGA	Used for preparing programs, data, etc. Word processors and text editors available on the market can also be used (those that support TEXT format) Macro assembler for 68000 Linker for 68000 Remote debugger for 68000


AUDIO OUTPUT SYSTEM
(Amp, Speakers, etc.)
HOST
COMPUTER
(Macintosh)
AUDIO OUT
SATURN TARGET BOX
(Target Board for sound
development)
MIDI IN
AUDIO IN (Digital)
RS-422
DIGITAL OUTPUT
MIDI OUT
MIDI INTERFACE
MIDI IN
DAT
CD PLAYER
MIDI OUT
MIDI KEYBOARD



2.0 Development Overview

The Saturn sound system makes it possible to produce sounds without having a knowledge of assembly language or other computer programming by comprising the target board itself as a MIDI-generated 32-voice polyphonic multiple sound source.

By creating tone bank data using a dedicated tone editing tool configured similarly to synthesizing tone editors available on the market, sound development can be performed to compose desktop music (DTM), and alike, using a regular MIDI sequencer. It is also possible to use a DSP linker to freely link reverb, delay, chorus and other effects (to the extent that memory will allow) generated by a third-party (off-the-shelf) digital multi-effector. Using the mixer function of a tone editing tool, it is possible to set the volume and type of the DSP effect together with the level or normal position of each tone in the tone bank data. Development of voice sampling can also be performed efficiently as the digital output of the target board can be input directly to the Macintosh where waveform editing is performed.

2.1 General Sound Development Procedure

1. Starting the Sound Simulator

When starting the sound simulator, the sound memory is mapped and the 68000 sound driver program is transferred to the target board. It is more convenient to prepare the sound area map in advance, but it can also be added to or changed as needed.

2. Starting the Tone Editor

Tone bank data are a compilation of mixer, voice, layer and waveform data, which make up one independent tone. Since voices correspond to program changes in MIDI, one bank can hold up to 128 voices. Therefore, up to 128 different types of musical instruments can be played with one bank of tone data.

Tone bank data (tone libraries to be supplied by SEGA, etc.) are transferred to the target board. At this stage, the tone editor functions as a multiple sound source that already generates sounds via the MIDI input. The key generation method used is DVA (last in receives priority). Tone, the level of each tone, /?pan?/, etc., can be edited and multiple tone bank data can be saved.

3. Starting the Waveform Editor

When editing the waveform itself (the basis of tone editing), start the waveform editor which is used to sample and edit the waveform.

4. Starting the DSP Linker

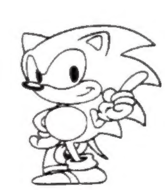
The DSP Linker is used to link DSP effect programs. After selecting the desired effect from the DSP library, the wiring is set and the DSP program for that effect is transferred to the target board. The various parameters of the effect can be edited after transfer. The number of voices that can be generated simultaneously for modulation effects are decreased from one to four voices since a slot is used as a modulator. See DSP editor chapter for details.

Since there are already reverb, echo, chorus and other module libraries, select and link the desired effect. Multiple effects can be used in the Saturn sound system, but you cannot exceed 128 steps overall. For example, suppose an echo was 20 steps, a chorus was 22 steps and an equalizer was 5 steps, then these three would total 47 steps.

Note: Steps are the number of commands in each effect.

5. Starting a Third-Party Sequencer

Composition and arrangements are performed using the target board as a sound source. The target board has two MIDI IN systems, and since each has 16 channels, up to 32 sequencer tracks can be accommodated. Since the VOICE number in the tone bank data can be freely selected and the tone changed in each sequence track by MIDI program changes, different tones can be set for all 32 tracks.



6. Converting Sequence Data

Composed tunes are ultimately converted to a MIDI standard file by the sequencer software (Performer, Vision, Cubase, etc.), and then it is converted by the sound simulator to Saturn format data for use with the target board.

There are two types of sequencer data: tune data produced by the MIDI sequencer and sequence data in the Saturn format, which is the same data compressed so that it can be loaded into the sound memory. Other than being compressed, the later is basically the same as developed MIDI data.

Note: Sound effects are composed basically by the same method as musical pieces. In this development system, there are no differences in the production process and parameter settings for musical compositions and sound effects.

7. Game Assembly Simulation (Hardware Simulation Function)

Saturn format sequence data are transferred to the target board. The sound simulator simulates tunes and sound effects as they would act in an actual game in order to check them. As long as there are no problems, the same sound that is generated by the target board with third-party software is reproduced. At this stage, the final evaluation of the links and balance between the tunes and sound effects is performed. If there are places that must be changed, each is redone using a tone editor, waveform editor, DSP linker or sequencer software.

8. The tunes and sound effects are then loaded into the actual game.

2.2 Composition of Tone Bank Data (Components of Tones)

Each component (beginning with the smallest) is explained below.

- **Waveform Data**

PCM data in AIFF format.

- **Layer Data (a layer is a single tone)**

The basic tone unit where LFO, EG, PITCH and FM settings and other data are added to the waveform data. Since volume can be controlled by the MIDI velocity value, velocity switches, etc., can be realized by setting a velocity table for each layer. One layer = one waveform. Up to 128 layers can be set in one voice.

- **Voice Data**

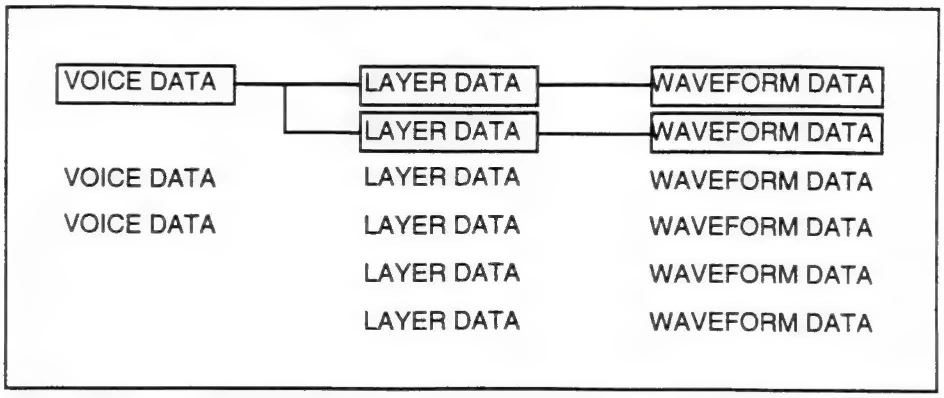
Voice data are data combining multiple layers to which keysplits, the volume for each layer, the bend range, the porutamento and other data are added. Up to 128 can be set in one bank. Voice data change according to the MIDI program changes.

Voice data comprise settings of how many of which layers and which layers will be sounded according to changes in the interval and volume.

- **Mixer Data**

In addition to the above data, the amount of return from DSP effects and pan (fixed position), orientation and other 16-channel mixing data can be set. Up to 128 can be set in one bank.

A combination of these is one tone bank data. One tone bank data has at least one voice and can have up to 128 voices, memory permitting. In this sound system, there may be several tone bank data in one map, and each can be sounded simultaneously as independent sound sources. Therefore, while a tune or sound effect is being played in one bank, another bank can be replaced. This makes it possible to realize a flexible system that facilitates efficient development and memory utilization.



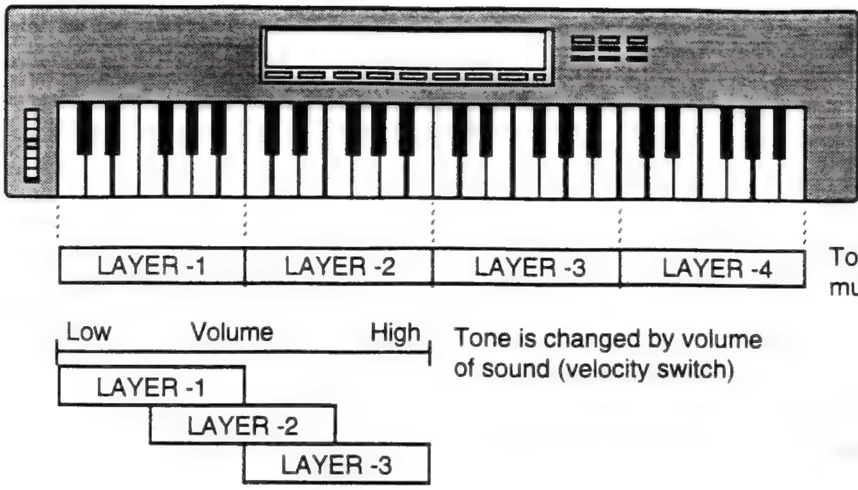
Note: Refer to tone editor for more information.

FM (Frequency Modulation)

FM has become well known through Yamaha's DX7 and other models, but the FM in this system is not limited to a sine wave as the fundamental waveform. The various AIFF format waveform data can be used as the carrier or the modulator. The algorithm for combining them can also be freely configured. It is also possible to change the degree of modulation by means of the velocity.

The biggest problem with FM is that the carrier and the modulator use one slot each, thus reducing the number of sounds that can be played simultaneously. However, since various tones can be produced by FM if a fundamental waveform (sine wave) is used as one waveform data, it is useful if you do not want to increase the tone bank data due to RAM restrictions.

(The memory for sound is 1024 Kbyte on the target board, but it is 512 Kbyte in the actual hardware. The area after subtracting sequence data, DSP programs, work RAM, etc., will be assigned to to tones. Even if the sampling rate is lowered, it is impractical to load several tens of types of PCM waveforms, which account for a large amount of data, into the memory at the same time.)



Tone is changed by musical interval (key split)

Tone is changed by volume of sound (velocity switch)

List of MIDI Commands that Can Be Used

Of the events contained in a standard MIDI file, those converted by the converter are listed below. As long as degradation storage of the pitch bend is not set in the user environment, the following can be converted with no degradation.

Note on/off	Note Off is replaced by Gate Time
Poly-Key Pressure	
Control Change	(Bank Select must exist at the top of each track)
Program Change	Must exit at the top of the track
Channel Pressure	
Pitch Wheel Change	Accommodates both 7bit and 14bit expressions
Meta Event	Accommodates only tempo

Note: System Message Exclusive, Start, Stop and Song Position are not accommodated.

If the following conditions are not satisfied, the converter outputs an error message and stops the conversion operation.

1. System messages must not be included in the standard MIDI file. System messages include Excusive, Start, Stop, Song, Position Pointer, Song Select, etc.
2. The number of events included in a standard MIDI file before conversion must be less than 6143 events per track. However, since Note Off is absorbed by Gate Time after conversion, the number of events becomes much smaller than this.

   Similarly, meta events must be less than 256, and each meta event must be not exceed 127 bytes in length.
3. Always convert standard MIDI files to type #1.
4. Only one loop start command (No. 31 of control changes) can exist in each track.
5. Always set a bank change and program change at the top of each track (not required in blank tracks). Voice numbers and bank numbers that do not exist cannot be designated by program change or bank select.

MIDI Channels and Voices

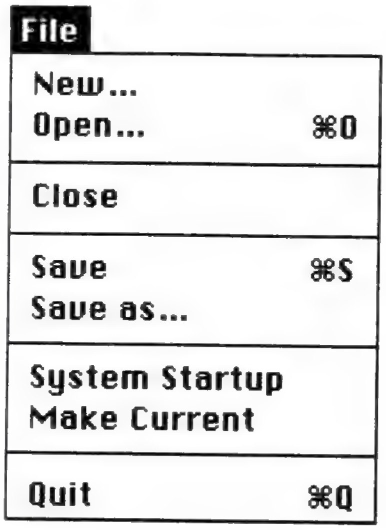
Since the Saturn sound system can handle MIDI data for up to 32 channels (tracks) simultaneously, up to 32 instruments can be handled when one MIDI channel is used per instrument. In other words, all of the songs and sound effects must be played (sounded) within this. There is no problem when a tune or sound effect is played independently. When any combination of a tune and a sound effect need to be played simultaneously, each must be assigned to a different channel. In this system, 32 MIDI channels are dynamically assigned (DVA) and sounds are generated.

Therefore, each sequence data must contain this information at the top of the track. In order to play the correct tone, the bank (bank select of control change) and voice (program change) must be designated for each track of sequence data.

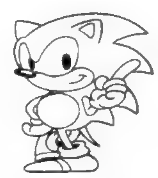
2.3 Using the Sound Simulator

There are five menus on the sound simulator menu bar.

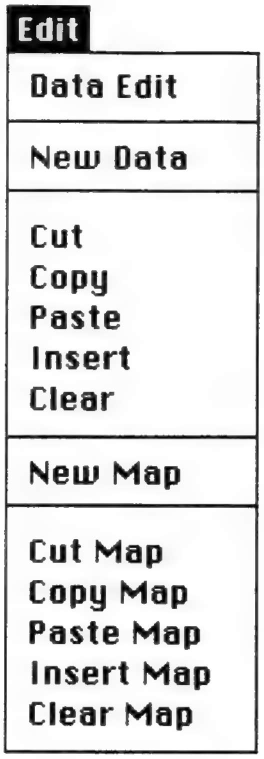
1. The **File** menu is shown below, and is used to open and save map files.



- A new map is made by selecting New.
- By selecting **System Startup,** sound drivers and other data are transferred to the target board and the sound system is started.
- The currently active map data are registered as the current map by selecting **Make Current**. This is convenient when you want to return to the current map by a simple operation even when opening multiple maps.



2. The Edit menu is shown below, and is used for data editing.



- New data can be registered by selecting **New Data**. After selecting the data to be registered from the selection box, input the start address and size.

- When **Cut** is selected, a confirmation menu appears in which you select whether or not to update the address after cutting. The cut data is stored to allow retrieval when **Paste** or **Insert** is selected.

- When **Copy** is selected, the currently selected data is stored and can be retrieved when **Paste** or **Insert** is selected.

- When **Paste** is selected, the currently stored data is pasted.

- When **Insert** is selected, the currently stored data is inserted immediately before the currently selected data, and is displayed inverted or at the end if no data is selected.

- When **Clear** is selected, all of the currently selected data, except the data type, are cleared.

Start	020000	Type	Tone Bank
Size	010000		Sequence Bank
File Name	Fighting 1		DSP Program
Auto Loader	☒		DSP Work

Cancel   Transfer   Save

- **New Map**, **Cut Map**, **Copy Map**, **Paste Map**, **Insert Map** and **Clear Map** perform similar editing on each piece of data as well as on the entire map.

3. The **Map** menu, shown below, is where maps are selected (transfer data to the target board), and is unlike the **Window** menu which only opens windows that display other maps. Up to 128 map data can be held.

Map

001 Opening
002 SCENE 1
003 SCENE 2
004 BOSS 1
005 SCENE 3

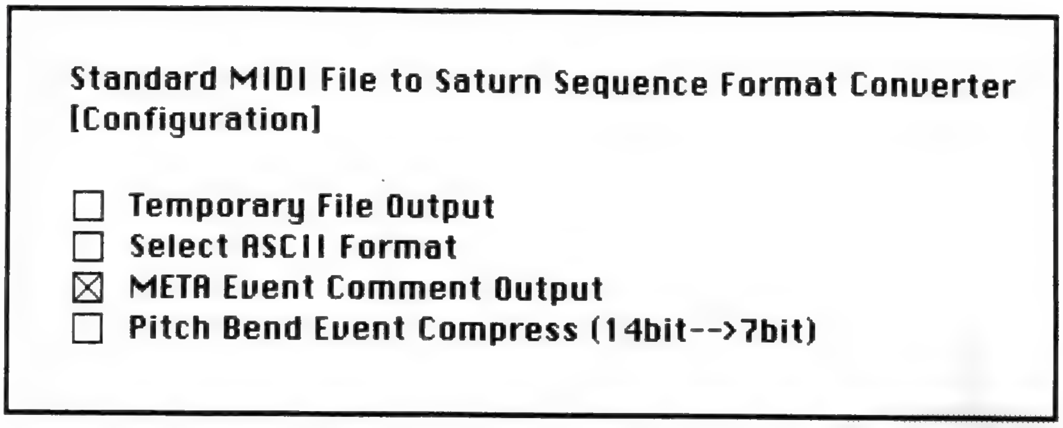
4. The **Window** menu, shown below, is used to switch between windows when referencing multiple maps. Up to 128 map data windows can be opened.

Window
All Close
Sound Simulator
001 Opening 002 SCENE 1 003 SCENE 2 004 BOSS 1 005 SCENE 3

5. The **Option** menu is shown below.

Option
Standard MIDI File Convert Converter Configuration Song Collect
Display Mode Option

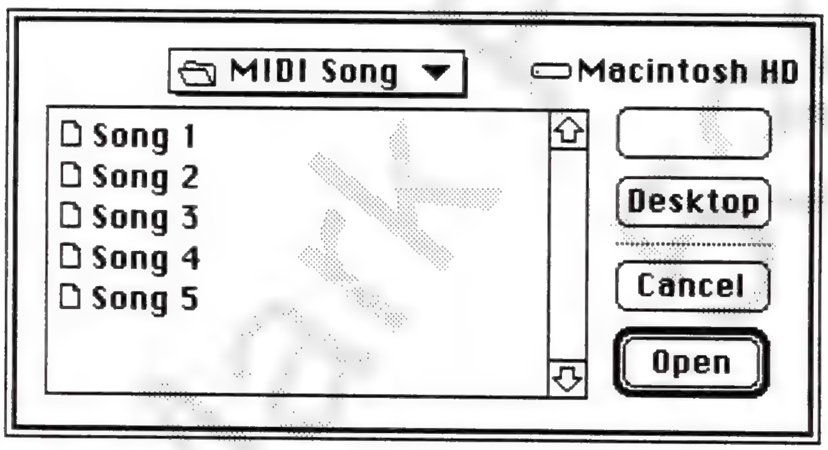
• When **Converter Configuration** is selected, the following menu is displayed.



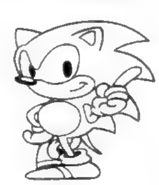
Each of the check boxes are explained below.

- **Temporary File Output**:	Selects whether the temporary file before repeated detection is output together or not.
- **Select ASCII File Format**:	Selects whether the final output file is in ASCII or binary format.
- **META Event Comment Output**:	Selects whether or not the META events are output as comments in an ASCII file.
- **Pitch Bend Event Compress**:	Selects whether the pitch bend is output in its original 14-bit precision or in com pressed 7-bit.

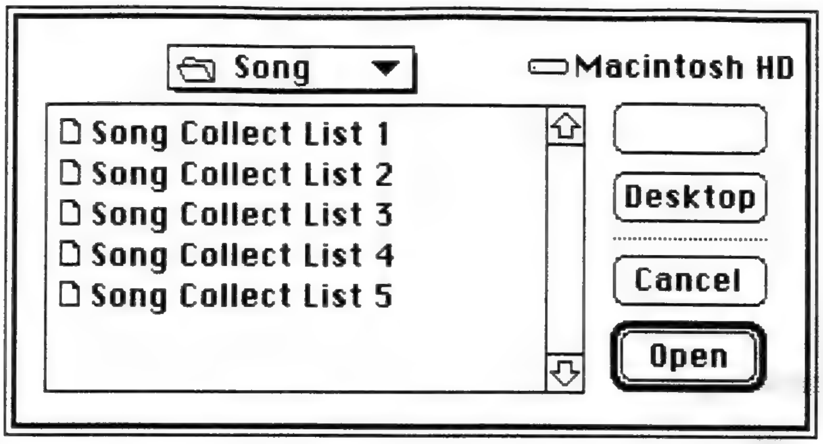
• When **Standard MIDI File Convert** is selected, the following dialog window is displayed showing the convertible standard MIDI files in the current directory.



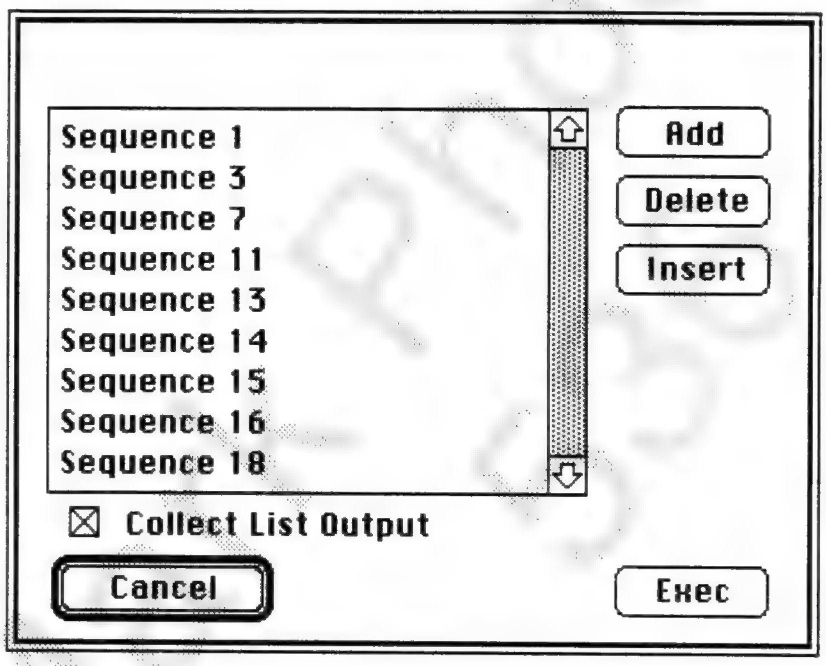
Conversion begins when the file has been selected, and the file is output in the Saturn's compressed format. The type of the output file and the compression mode are specified in **Converter Configuration**. The output file is listed with the extension ".CNV"; temporary files are output with the extension ".TMP" on the file name.



- When **Song Collect** is selected, the following dialog box appears.



If there is a collect list file for Song Collect, then that is read in. The collect list is a compilation of several song names and makes selection of song names easier.
If there is no collect list file, then it must be made by pressing the **Exec** button to begin song collection.



Add	Adds to the list
Delete	Deletes a list
Insert	Inserts a list

2.4 MIDI Specification

```
1:TRANSMITTED DATA
            1-1:CHANNEL MESSAGES
                        transmit data nothing

            1-2:SYSTEM MESSAGES
                        transmit data nothing

            1-3:SYSTEM EXCLUSIVE MESSAGES
                        Refer to "MIDI EXCLUSIVE"
2:RECOGNIZED RECEIVE DATA

            2-1:CHANNEL MESSAGES
      <<<< note off >>>>
      status    second     third          description
      1000 nnnn  0kkk kkkk  0xxx xxxx  note off

      <<<< note on >>>>
      status    second     third          description
      1001 nnnn  0kkk kkkk  0000 0000  note off
      1001 nnnn  0kkk kkkk  0vvv vvvv  note on
                                       vvv vvvv = 1 - 127

      <<<< control change >>>>
      status    second     third          description
      1011 nnnn  0000 0000  0vvv vvvv  bank change# LSB
      1011 nnnn  0000 0001  0vvv vvvv  modulation
      1011 nnnn  0000 0101  0vvv vvvv  portament time
      1011 nnnn  0000 0111  0vvv vvvv  master volume
      1011 nnnn  0000 1010  000v vvvv  pan pot
      1011 nnnn  0001 0000  0vvv vvvv  mixer change
      1011 nnnn  0001 0001  0vvv vvvv  effect pan pot
      1011 nnnn  0010 0000  0vvv vvvv  bank change# MSB
      1011 nnnn  0100 0000  0vvv vvvv  damper
      1011 nnnn  0100 0001  0vvv vvvv  portament off
                                         vvv vvvv = 0 - 63
      1011 nnnn  0100 0001  0vvv vvvv  portament on
                                         vvv vvvv = 64 - 127
      1011 nnnn  0101 1011  0vvv vvvv  effect change
      1011 nnnn  0111 1011  0000 0000  all note off
```

```
<<<< program change >>>>
status     second      third          description
1100 nnnn  0ppp pppp   — —   voice change

<<<< program change >>>>
status     second      third          description
1110 nnnn  0bbb bbbb 0bbb bbbb  bender change
```

3:SYSTEM MESSAGES

```
1111 0000  Exclusive message        Refer to MIDI EXCLUSIVE
1111 0001  quater flame Message     Ignore
1111 0010  song position data       Ignore
1111 0011  song select              Ignore
1111 0110  tune request             Ignore
1111 0111  End of exclusive         Refer to MIDI EXCLUSIVE
1111 1000  timing clock             Ignore
1111 1010  start                    Ignore
1111 1011  continue                 Ignore
1111 1100  stop                     Ignore
1111 1110  active sensing           Ignore
1111 1111  system reset             Used
```

MIDI EXCLUSIVE

SATURN SYSTEM EXCLUSIVE

```
1st Byte = 1111 0000 (F0H) : Exclusive Status
2nd Byte = 0100 0011 (43H) : YAMAHA ID
3rd Byte = 0111 1001 (79H) : DIV               Exclusive
4th Byte = 0iii iiii (iiH) : Device ID           Header
5th Byte = 0000 0001 (01H) : Saturn ID
6th Byte = 0fff ffff (ffH) : Command Code
7th Byte = 0ddd dddd (ddH) : data
  :            :           :  :
LastByte = 1111 0111 (F7H) : End of Exclusive
```

Note: The device ID is for identification when multiple MIDI devices are connected to the Saturn Development Board. Please set to "0" in general.

Command Code List

Command	Description	Receive	Ttansmit
00H	data Dump Request	o	
01H	data Set	o	
02H	Acknowledge	o	o
03H	Not Acknowledge	o	o
04H	Reset	o	
05H	SCSCBIN data change	o	
41H	data Dump		o

Note:

Receive : SATURN Target BOARD EXT.MIDI Instrument
Transmit : SATURN Target BOARD EXT.MIDI Instrument

MIDI EXCLUSIVE FORMAT

R: receive; T: transmit

Command=00H : data Dump Request R

Byte	Description
F0H, 43H, 79H, iiH, 01H	Exclusive Headder
0000 0000	data Dump Request
0000 mmmm	byte size (see Note 1-1)
0000 mmmm	
0000 kkkk	start address (see Note 1-2)
0000 kkkk	
0000 kkkk	
0000 kkkk	
0000 kkkk	
0000 kkkk	
0sss ssss	check sum (see Note 1-3)
F7H	EOX

Receive this message, and transmits Command = 41H message or Command = 03H message.

```
Command=01H : data Set                                    R
     Byte                     Description
 F0H, 43H, 79H, iiH, 01H  Exclusive Header
 0000 0001                data Set
 0000 mmmm                byte size     (see Note 1-1)
 0000 mmmm
 0000 kkkk                start address (see Note 1-2)
 0000 kkkk
 0000 kkkk
 0000 kkkk
 0000 kkkk
 0000 kkkk
 0000 hhhh                write data    (see Note 1-4)
 0000 1111
     ~
 0000 hhhh
 0000 1111
 0sss ssss                check sum     (see Note 1-3)
 F7H                      EOX
 Receive this message, and transmits Command = 02H message
 or Command = 03H message.
```

```
Command=02H : Acknowledge                                 R, T
     Byte                     Description
 F0H, 43H, 79H, iiH, 01H  Exclusive Header
 0000 0010                Acknowledge
 F7H                      EOX
 Transmits this message when processing completed.
```

```
Command=03H : Not Acknowledge                             R, T
     Byte                     Description
 F0H, 43H, 79H, iiH, 01H  Exclusive Header
 0000 0011                Not Acknowledge
 F7H                      EOX
 Transmits this message when processing error.
```

Command=04H : Reset R

Byte	Description
F0H, 43H, 79H, iiH, 01H	Exclusive Header
0000 0100	Reset
F7H	EOX

Transmits this message when SATURN reset.

Command=05H : Reset R

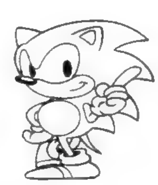
Byte	Description	
F0H, 43H, 79H, iiH, 01H	Exclusive Headder	
0000 0101	SCSPBIN data change	
0000 cccc	channel#	(see Note 1-5)
0000 0aaa	mode#	(see Note 1-7)
0000 dddd	Number	(see Note 1-6)
0000 dddd		
00vv vvvv	select#	(see Note 1-8)
0000 hhhh	write data	(see Note 1-4)
0000 1111		
0sss ssss	check sum	(see Note 1-3)
F7H	EOX	

Receive this message, and transmits Command = 02H message or Command = 03H message.

Command=41H : data Dump T

Byte	Description	
F0H, 43H, 79H, iiH, 01H	Exclusive Header	
0100 0001	data Dump	
0000 mmmm	byte size	(see Note 1-1)
0000 mmmm		
0000 kkkk	start address	(see Note 1-2)
0000 kkkk		
0000 kkkk		
0000 kkkk		
0000 kkkk		
0000 kkkk		
0000 hhhh	Dump data	(see Note 1-4)
0000 1111		

~



```
0000 hhhh
   0000 1111
   0sss ssss                check sum     (see Note 1-3)
   F7H                      EOX
   Receive this message, and transmits Command = 02H
   message or Command = 03H message.
```

Note 1-1 mmmm mmmm = byte size (01H ~ FFH)

Note 1-2 kkkk...kkkk = SATURN memory Address (000000H~FFFFFFH)

Note 1-3 sss ssss = check sum (00H ~ 7FH)
7th byte + 8th byte + ... + check sum = 000 0000B

Note 1-4 hhhh llll = byte data (00H ~ FFH)
write/Dump data

Note 1-5 cccc = channel number (00H ~ 0FH)

Note 1-6 dddd dddd = MIXER, VL, PEG, PLFO, VOICE or LAYER Number

Note 1-7 aaa = mode# (see Note 2-1)

- 0: MIXER change
- 1: VL change
- 2: PEG change
- 3: PLFO change
- 4: VOICE change
- 5: LAYER change

Note 1-8 vv vvvv = select# (see Note 2-1)

Note 2-1 mode#-select# table

mode# = 0 (MIXER change)					
select#	00H	EFSDL	[EFREG0]		
	01H	EFPAN	[//]		
	02H	EFSDL	[EFREG1]		
	03H	EFPAN	[//]		
	-	-			
	1EH	EFSDL	[EFREG15]		
	1FH	EFPAN	[//]		
	20H	EFSDL	[EXTS0]		
	21H	EFPAN	[EXTS1]		

mode# = 4 (VOICE change)		
select#	00H	Play Mode
	01H	Bend Range
	02H	Portament time
	03H	Layer Number-1
	04H	Volume Bias
mode# = 5 (LAYER change)		
select#	00H	start MIDI note
	01H	end MIDI note
	02H	PEON

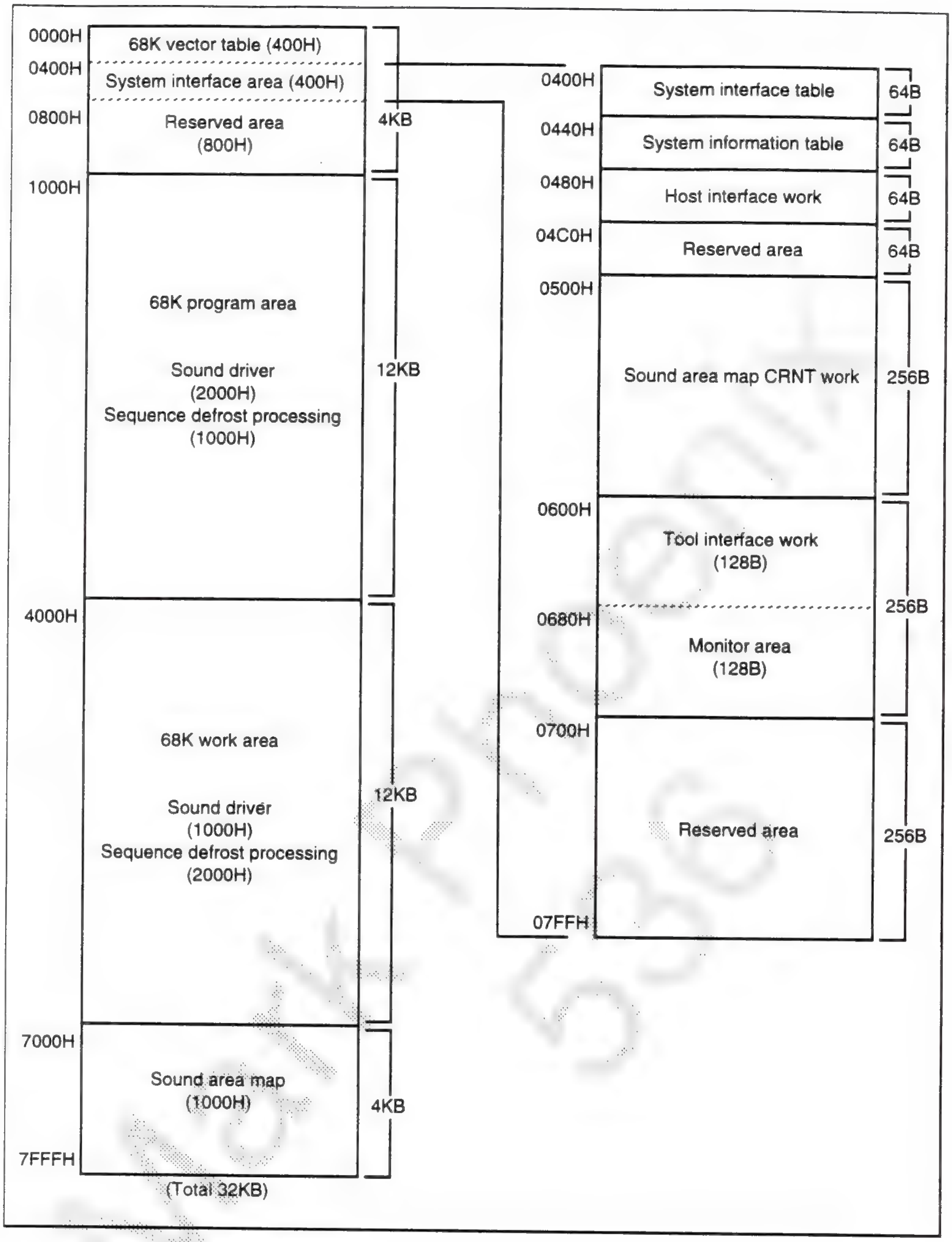
```
 mode# = 1 (VL change)
select#  00H  encoded-Rate0
         01H  Velocity-Point0
         02H  Velocity-Level0
         03H  encoded-Rate1
         04H  Velocity-Point1
         05H  Velocity-Level1
         06H  encoded-Rate2
         07H  Velocity-Point2
         08H  Velocity-Level2
         09H  encoded-Rate3
 mode# = 2 (PEG change)
select#  00H  DLY
         01H  OL
         02H  AR
         03H  AL
         04H  DR
         05H  DL
         06H  SR
         07H  SL
         08H  RR
         09H  RL
 mode# = 3 (PLFO change)
select#  00H  DLY
         01H  FRQR
         02H  HT
         03H  FDCT
```

```
03H  PLON
04H  D2R
05H  D1R
06H  AR
07H  KRS
08H  DL
09H  RR
0AH  Mod.Wheel
0BH  TL
0CH  MDL
0DH  LFOF
0EH  PLFOWS
0FH  PLFOS
10H  ALFOWS
11H  ALFOS
12H  ISEL
13H  IMXL
14H  DISDL
15H  DIPAN
16H  BASE note
17H  Fine tune
18H  GN & Layer#
19H  GN & Layer#
1AH  VL#
1BH  PEG#
1CH  PLFO#
1DH  reserved
1EH  reserved
1FH  reserved
```

Specification submitted to the MIDI Association (February 28, 1994).

Chapter 2:
System Interface

1.0 System Area



2.0 Contents of System Area

The system area is a fixed area for running the sound driver offered by SEGA, and cannot be used for any other application. Since the mapping of this area cannot be changed, the user need not be concerned about the contents of this area. The 32 Kbyte from the top (00000H to 8000H) of the 4Mbit of sound memory is used, and the contents of this area is described below.

- 68K vector table
  The vector table for program interrupt processing by the sound CPU (68K). The size is fixed at 400H and cannot be changed.

- System Interface Area
  A fixed area for interfacing between sound drivers, tone development systems, the host system for game assembly, etc., and this system. It extends from 0400H to 400H and cannot be changed.

- 68K Program Area
  This is the program area for the sound CPU and is used to store and execute all programs related to sound. The top address and size of this area are stored in the system information table of the system interface area.

- 68K Work Area
  This is the program work area for the sound CPU and is used as a work area by all sound-related programs. The top address and size of this area are stored in the system information table of the system interface area.

- Sound Area Map
  The sound area map is stored here. Up to 128 area maps can be held in one sound area map (one area map can hold up to 32 map data). Using a sound simulator, one sound area map can be made for one game. Since this area is only for storing the entire sound area map, the map data of the currently selected area references sound area map CRNT work of the system interface area. The top address and area size are stored in the system information table in the system interface area.

3.0 System Interface Area

In the Saturn sound system, a system interface area provided in the fixed area of the sound memory is used to exchange information between sound drivers, tone development systems, the host system for game assembly and other systems. It comprises an information table that stores system information and a work area for exchanging information.

3.1 System Interface Table (400H–43FH: 64B)

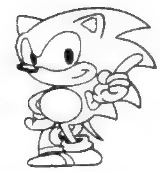
This is an information table for controlling the interface between each of the systems during sound development or game assembly and is stored at a fixed address in the sound memory. It also contains work, etc., for the BOOT ROM during sound development.

Address	Offset	Size	Area	Contents
0400	+00	4B	System information table pointer	Top address of system information table (0440H)
0404	+04	4B	Host interface work pointer	Top address of host interface work (0480H)
0408	+08	4B	Sound area map CRNT work pointer	Top address of sound area map CRNT work (0500H)
040C	+0C	4B	Tool interface work pointer	Top address of sound tool interface work (0600H)
0410	+10	1B	DSP Program load flag	BOOT ROM program work
0411–043F	–	47B	–	Reserved area

3.2 System Information Table (440H–47FH: 64B)

This is an information table in which system information of the sound system is stored at a fixed address in the sound memory.

Address	Offset	Size	Information Data	Contents
0440	+00	4B	68K program start address	Top address of 68K program area (1000H)
0444	+04	4B	68K program size	Size of 68K program area (3000H)
0448	+08	4B	Sound area map start address	Top address of sound area map area (7000H)
044C	+0C	4B	Sound area map size	Size of sound area map area (1000H)
0450	+10	4B	68K work start address	Top address of 68K work area (4000H)
0454	+14	4B	68K work size	Size of 68K work area (3000H)
0458 - 047F	–	40B	–	Reserved area



3.3 Host Interface Work (480H–4BFH: 64B)

This is a work area for communicating between the host system and the sound system. Commands are received in this area from the host system, and returns status/timing flags, etc. Basic control is performed in the areas listed below, but since specific commands are required, depending on the project (game), unused area is allotted as required. Modes, status, etc., can similarly be added or changed as required.

pointer + xx	Size	Interface Data	Contents
+ 00 (hex)	2B	Command Data	Command from host to sound
+ 02	2B	Parameter Data 1	Command parameter 1
+ 04	2B	Parameter Data 2	Command parameter 2
+ 06	2B	Parameter Data 3	Command parameter 3
+ 08	2B	Parameter Data 4	Command parameter 4
+ 0A	2B	Parameter Data 5	Command parameter 5
+ 0C	2B	Parameter Data 6	Command parameter 6
+ 0E	2B	Parameter Data 7	Command parameter 7
+ 10	2B	Song 1 mode/status	Bits 8-15 (high byte): song 1 mode Bits 0- 7 (low byte): song 1 status
+ 12	2B	Song 2 mode/status	Bits 8-15 (high byte): song 2 mode Bits 0- 7 (low byte): song 2 status
+ 14	2B	Song 3 mode/status	Bits 8-15 (high byte): song 3 mode Bits 0- 7 (low byte): song 3 status
+ 16	2B	Song 4 mode/status	Bits 8-15 (high byte): song 4 mode Bits 0- 7 (low byte): song 4 status
+ 18	2B	Song 5 mode/status	Bits 8-15 (high byte): song 5 mode Bits 0- 7 (low byte): song 5 status
+ 1A	2B	Song 6 mode/status	Bits 8-15 (high byte): song 6 mode Bits 0- 7 (low byte): song 6 status
+ 1C	2B	Song 7 mode/status	Bits 8-15 (high byte): song 7 mode Bits 0- 7 (low byte): song 7 status
+ 1E	2B	Song 8 mode/status	Bits 8-15 (high byte): song 8 mode Bits 0- 7 (low byte): song 8 status
+ 20 - 3F	32B	Reserved	–

```
[song mode]
00:   Stop
01:   Play
02:   Fade in
03:   Fade out
04:   Play pause
05:   Fade in pause
06:   Fade out pause
```

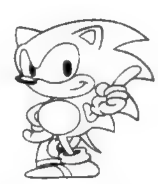
```
[song status]
00:          normal
01-7F:       error code
80-FF:       timing flag
```

Note: Use caution since the sound CPU cannot operate while the host system is accessing the sound memory. Keep reading and writing of the sound memory by the host system to a minimum and do not allow continuous access over long periods.

3.4 Sound Control Command

The table below describes the commands and parameters issued to the sound system from the host system. Basic control is performed using these commands, but since special commands become necessary depending on the project (game), unused command codes are allotted as required. Commands and parameters can similarly be added or changed as required.

Command Name	Command Data	Parameter Data
Reserved	00 (hex)	Nothing
Sequence Start	01	P1 0-7: sound control number P2 0-15: sequence bank number P3 0-255: sequence song number P4 0-7: priority level
Sequence Stop	02	P1 0-7: sound control number
Sequence Pause	03	P1 0-7: sound control number
Sequence Continue	04	P1 0-7: sound control number
Fade In	05	P1 0-7: sound control number P2 0-15: sequence bank number P3 0-255: sequence song number P4 0-7: priority level P5 0-255: Maximum speed at fade level 0
Fade Out	06	P1 0-7: sound control number P2 0-255: Fade level; maximum speed at 0
Tempo Change	07	P1 0-7: sound control number P2 +32767 --> -32768: tempo value; value relative to current value
Map Change	08	P1 0-255: area number of sound area map that can be changed
MIDI Direct Control	09	P1 0-31: track (channel) number P2 0-255: MIDI command P3 0-127: MIDI data 1 P4 0-127: MIDI data 2
CD–DA Level	80	P1 0-7: left; 8 steps, 0 is OFF P2 0-7: right; 8 steps, 0 is OFF
CD–DA Pan	81	P1 0-31: left; 32 steps P2 0-31: right; 32 steps
Total Volume	82	P1 0-15: 16 steps, 0 is OFF
Effect Change	83	P1 0-15: effect bank number
Reserved	84-FF	Nothing



Sound Control Number

Up to eight sequences can be controlled when sequences are played, and are specified by sound control numbers 0-7 when the sequences are started (fade in). Subsequent stop, pause, continue and other commands are executed according to these sound control numbers.

Sequence Bank Number

When multiple sequence data banks where sequence data is stored are mapped in the currently active map, this number specifies the number of the sequence data bank in the active map. Since a maximum of 16 of the same banks can be held in one map, this number ranges from 0 to 15. If there is only one sequence data bank, this number is always 0.

Sequence Song Number

Specifies the number of the sequence data in the sequence data bank when there are multiple numbers of sequence data stored. Up to 256 sequences can be stored as long as there is enough disk space.

Priority Level

Specifies the priority level in eight grades when sequences are played. The highest priority is assigned "0", the lowest is "7". Sequences of the same priority level are all played simultaneously, but sequences with lower priority are not played until higher priority levels are done.

Fade Level

The time when playing begins until maximum volume during fade-in is specified in 256 steps (0-255). At "0" (the highest speed), the maximum volume is reached when playing begins. The time from when playing stops until minimum volume during fade-out is specified in 256 steps (0-255). At "0", the minimum volume is reached when playing stops.

Effect Bank Number

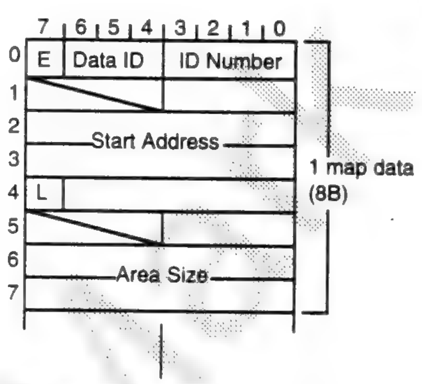
This number specifies the number of each DSP program bank in the active map. Since up to 16 banks can be held in one map, the number ranges from 0 to 15. When there is only one sequence data bank, this number is always "0".

Sound Area Map CRNT Work (500H-5FFH: 256B)

The area map (up to 32 map data called the active map) of the currently selected area is stored in this area. When a map change is received from the host system, the sound system transfers the area map data of the active area to this area. One map data comprises 8 bytes, and one area can hold up to 32 map data (8 x 32 = 256B). Since the map data are stored in random order and the number of map data can vary, data are searched by type and data number. Data are stored from the top, and when there is no more data, the data end bit of the map data is set.

pointer + xx	Size	Interface Data	Contents		
+ 00 (hex)	1B	End mark Data ID ID number	Data end bit Data type Data number	(bit 7) (bits 4-6) (bits 0-3)	0/1 0-3 0-15
+ 01	3B	Start address	Area start address	(bits 0-19)	00000-FFFFF
+ 04	1B	Load mark	Transfer end bit	(bit 7)	0/1
+ 05	3B	Area size	Area size	(bits 0-19)	00000-FFFFF
|					
+ F8	1B	End mark Data ID ID number	Data end bit Data type Data number	(bit 7) (bits 4-6) (bits 0-3)	0/1 0-3 0-15
+ F9	3B	Start address	Area start address	(bits 0-19)	00000-FFFFF
+ FC	1B	Load mark	Transfer end bit	(bit 7)	0/1
+ FD	3B	Area size	Area size	(bits 0-19)	00000-FFFFF

Bit Image of Map Data



E	0: data exists 1: no data
Data ID	0: tone bank data 1: sequence data 2: DSP program 3: DSP work RAM
ID Number	0-15: designation numbers in data ID
L	0: transfer data end 1: transfer complete

Tool Interface Work (600H–6FFH: 256B)

Work that stores RAM area Information used by the waveform editor, tone editor, DSP linker. The lower 128 bytes is a monitor data area for monitoring the play status.

pointer + xx	Size	Data	Contents
+ 00 (hex)	2B	–	Reserved
+ 02	2B	–	Reserved
+ 04	4B	Area Start Address	Start address of waveform editor RAM area
+ 08	4B	Area total size	Size of waveform editor RAM area
+ 0C	2B	–	Reserved
+ 0E	4B	Area Start Address	Start address of tone editor RAM area
+ 12	4B	Area total size	Size of tone editor RAM area
+ 16	2B	–	Reserved
+ 18	4B	TrgtMem_DSPprogAddress	DSP linker dedicated area
+ 1C	4B	TrgtMem_DSPprogSize	DSP linker dedicated area
+ 20	32B	TrgtMem_Filename	DSP linker dedicated area
+ 40	4B	TrgtMem_DSPRAMSize	DSP linker dedicated area
+ 44	2B	TrgtMem_RBL	DSP linker dedicated area
+ 46 - 7F	60B	–	Reserved
+ 80 - 83	4B	Voice 1 monitor	Bits 24-31 (1st byte): Program (voice) number 0-127 Bits 16-23 (2nd byte): MIDI note number 0-127 Bits 08-15 (3rd byte): MIDI velocity 0-127 Bits 00-07 (4th byte): Reserved
+ FC - FF	4B	Voice 32 monitor	Bits 24-31 (1st byte): Program (voice) number 0-127 Bits 16-23 (2nd byte): MIDI note number 0-127 Bits 08-15 (3rd byte): MIDI velocity 0-127 Bits 00-07 (4th byte): Reserved

4.0 Host Interface

Since the sound system runs in RAM, all data, including the sound system program, are lost when the power is turned off. Therefore, the sound system must be restarted each time the power is turned on. When developing tones, the sound simulator, not the host system, initializes the sound system. But when assembling games, the host-side system must do all initialization. Also, the sound CPU is reset when the system power is turned on and cannot operate until the host system releases this reset. Therefore, the sound system should be started as described below.

4.1 Starting the Sound System

1. Initialize the SCSP registers so that the SCSP can access the sound memory.
2. Transfer the 128 byte information table part of the system interface table to a fixed area (400h-47Fh).
3. Transfer the sound program to the sound memory. Reference the system interface table for the destination and size.
4. Transfer the sound area map to the sound memory. Reference the system interface table for the destination and size.
5. Set the reset vector for the sound CPU. Reference the system interface table for the address. SSP does not need to be set.
6. Cancel reset of the sound CPU.

This operation will start the sound CPU and run the program from the reset vector in the sound memory. The SCSP registers, and canceling reset and other hardware subjects are described in greater detail in the SCSP User's Manual.

4.2 Preparing for Play

The sound controller is prepared by starting the sound system; preparation for play is performed next. Because transfer of tune data and sound effect data (referred to as sequence data together) to the sound memory and playing of tunes and sound effects are all performed according to the sound area map, the initial area is first designated. When ever there is a subsequent change in the area, the map change command must be implemented for the area to be changed.

1. Issue the map change command for the initial area.
2. Transfer the tone data to the sound memory. Reference sound area map CRNT work for destination and size.
3. Transfer sequence data to the sound memory. Reference sound area map CRNT work for destination and size.
4. Transfer the DSP program to the sound memory. Reference sound area map CRNT work for destination and size.
5. Control the start/stop, etc., of the sequence data.

4.3 Sound Control

Sound control of the start/stop, etc., of the sequence data is performed by command communication with the host system which uses host interface work. Refer to "Command Interface" in this manual for more information on communication commands, modes and status.

4.4 Hardware Interface

Communication between the host and the sound interface can be performed by two methods. One, which uses the sound memory, is command handshake which utilizes the ability of the host system to access (R/W) the sound memory. Method two communicates using interrupts, wherein the host is connected to the sound system by two signals and the sound system is connected to the host system by one signal. In this version, which allows selection of the optimum system depending on the game content or system, sound memory is used.

4.5 Software Interface

The host system and the sound system are connected via SCU, but because access (R/W) of the memory is one way from the host to the sound memory, the communication protocol is expressed by a handshake that uses the host interface work. The host system confirms that the command data area of the host interface work is 00H and writes the command and necessary parameters, and the sound system initializes the command data area at 00H when it receives the command. Since the sound system continually updates the song mode/song status area regarding the play status, etc., of each sequence, have the host system reference (read) this area.

The CD-BIOS performs all control for the CD-DA and sends the sounds to the sound system. The sound system can change the volume and the left/right pan of the sound and apply effects. In order to apply effects to the CD-DA, a separate DSP program is required. Normally, effects are altered through changes in the sequence data, but please perform effect changes from the host should it be necessary.

SEGA OF AMERICA, INC.
Consumer Products Division

SATURN Sound Simulator Manual

Doc. # ST-168-R3-011895

General Notice

When using this document, keep the following in mind:

1. This document is confidential. By accepting this document you acknowledge that you are bound by the terms set forth in the nondisclosure and confidentiality agreement signed separately and in the possession of SEGA. If you have not signed such a nondisclosure agreement, please contact SEGA immediately and return this document to SEGA.

2. This document may include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new versions of the document. SEGA may make improvements and/or changes in the product(s) and/or the program(s) described in this document at any time.

3. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without SEGA's written permission. Request for copies of this document and for technical information about SEGA products must be made to your authorized SEGA Technical Services representative.

4. No license is granted by implication or otherwise under any patents, copyrights, trademarks, or other intellectual property rights of SEGA Enterprses, Ltd., SEGA of America, Inc., or any third party.

5. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein.

6. It is possible that this document may contain reference to, or information about, SEGA products (development hardware/software) or services that are not provided in countries other than Japan. Such references/information must not be construed to mean that SEGA intends to provide such SEGA products or services in countries other than Japan. Any reference of a SEGA licensed product/program in this document is not intended to state or simply that you can use only SEGA's licensed products/programs. Any functionally equivalent hardware/software can be used instead.

7. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

> NOTE: A reader's comment/correction form is provided with this document. Please address comments to :
>
> SEGA of America, Inc., Developer Technical Support (att. Evelyn Merritt)
> 150 Shoreline Drive, Redwood City, CA 94065
>
> SEGA may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

(11/2/94- 002)

REFERENCES

In translating/creating this document, certain technical words and/or phrases were interpreted with the assistance of the technical literature listed below.

1. *Dictionary of Science and Engineering, 350,000 words, 3rd Edition*
   Inter Press
   Tokyo, Japan
   1990

2. *Computer Dictionary*
   Kyoritsu Publishing Co., LTD.
   Tokyo, Japan
   1978

3. *IBM Dictionary of Computing*
   McGraw-Hill, Inc.
   New York, New York
   1994

READER CORRECTION/COMMENT SHEET

Keep us updated!

If you should come across any incorrect or outdated information while reading through the attached document, or come up with any questions or comments, please let us know so that we can make the required changes in subsequent revisions. Simply fill out all information below and return this form to the Developer Technical Support Manager at the address below. Please make more copies of this form if more space is needed. Thank you.

General Information:

Your Name ____________________ **Phone** __________

Document number ST-168-R3-011895 **Date** __________

Document name SATURN Sound Simulator Manual

Corrections:

Chpt.	pg. #	Correction

Questions/comments: ____________________

Where to send your corrections:

Fax: (415) 802-1717
Attn: Evelyn Merritt,
Developer Technical Support

Mail: SEGA OF AMERICA
Attn: Evelyn Merritt,
Developer Technical Support
150 Shoreline Dr.
Redwood City, CA 94065

The Sound Simulator

Main Window

Edit window (samplemap)

CRNT 0 No name

No		Start - End	Size	Data	File name
01x	L	10000-1FFFF	10000	Sequence1	mureseq
02		20000-3FFFF	20000	DPS WorkRAM1	
03x	L	40000-BFFFF	80000	BANK data1	Apus area1 set.bin
04		C0000-CFFFF	10000	Sequence2	mureseq
05		D0000-DFFFF	10000	Sequence3	NoName
06		E0000-E07FF	00800	DPS program1	dsp1.EXB
07		E0800-E0FFF	00800	DPS program2	dsp2.EXB
08		E1000-E17FF	00800	DPS program3	

Sound Simulator

The Sound Simulator is a simulation tool that enables the playback of sound data as it should sound when incorporated into a game. Playback control of sound data is typically not possible until the data is paired with the game application code. The Sound Simulator enables the playback of sound data by simulating the game code on the Macintosh. The Sound Simulator's functions are as follows:

- Sound system startup
- Tone data and song data transfer
- Sound playback control
- DSP (effects) program change
- Creation, compression, and linking of SATURN format data
- Creation of sound area maps

Sound System Start Up

When the power to the sound board is turned on, the sound operating system must be started up. As with the production SATURN, the sound operating system is cleared from memory at power-down. Click on **System Startup** to perform the following processes from the Sound Simulator.

- Hardware initialization
- System table and sound area map transfer
- Sound program transfer
- Sound driver startup

Tone Data and Song Data Transfer

Tone data and song data must be transferred to the sound memory according to a sound area map before any sound can be played back. If the tone data and song data are within the area sizes of the map data, those data can be changed dynamically. Therefore, several tone data and song data areas can be used. Song data can be loaded in while a song is played back.

Sound Playback Control

Start, stop, pause, fade in, fade out, etc. for songs and sound effects can be controlled from the Sound Simulator. In addition to the mouse, function keys can be assigned to the Macintosh keyboard to control real-time playback. Evaluation of sound effects and level matching can be performed while a song is playing.

DSP (Effect) Program Change

When there are several DSP (effect) programs on the sound area map, a DSP program can be changed by clicking on the **Effect Change** button. The DSP program does not run by simply being transferred into memory, so even if there is only one DSP program, **Effect Change** still must be selected.

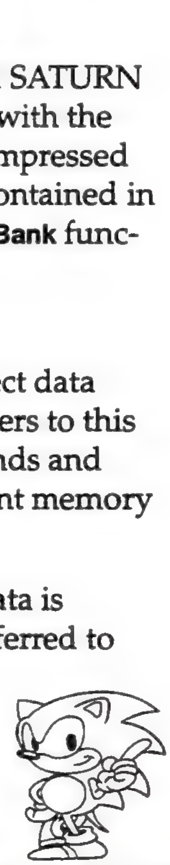
Creation, Compression, and Linking of SATURN Format Data

Song data created with a MIDI sequencer can be converted to compressed SATURN data. Multiple songs can be put into one song data bank (sequence bank) with the **Make Sequence Bank** function. This function links together independent compressed data files. It is assumed that multiple song (and/or sound effect) data is contained in a sequence bank, so even if only one song is in a bank, the **Make Sequence Bank** function must be used.

Creation of Sound Area Maps

Sound is controlled by each individual game area. The tone, song and effect data size of each area are used to create a memory map. The game program refers to this map and transfers tone and song data; the sound driver also controls sounds and playback based on this map. Therefore, the sound area map is an important memory map that is at the heart of sound control.

A unique sound area map is made for each game. The sound area map data is incorporated in the game program by the programmer. This data is transferred to the sound system when the game application is started up.



Using the Sound Simulator

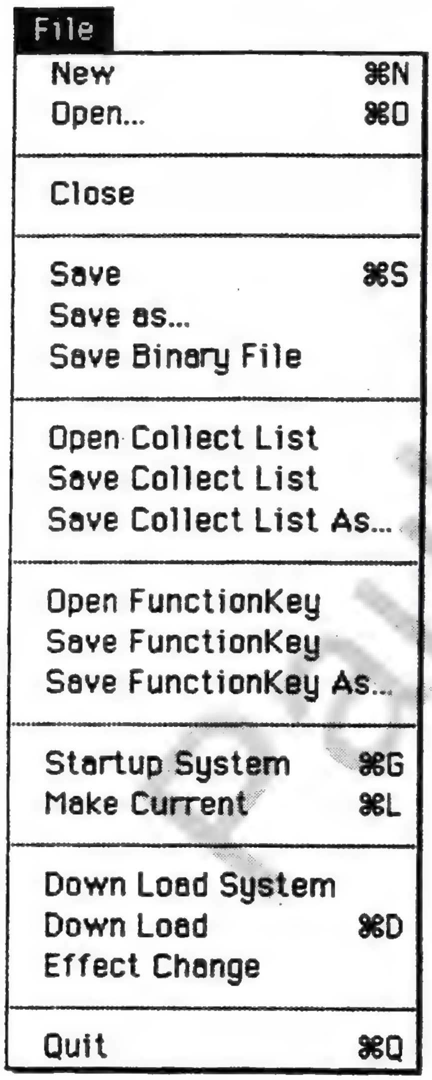
This section explains the functions for all Sound Simulator menu items and their uses.

Menu Bar

There are five menus on the Sound Simulator menu bar.

File Menu

Selecting this menu displays the following:



The File menu is the menu that is used to open, save sound area map files, etc.

New
Creates a new map.

Open...
Opens a map that has already been created.

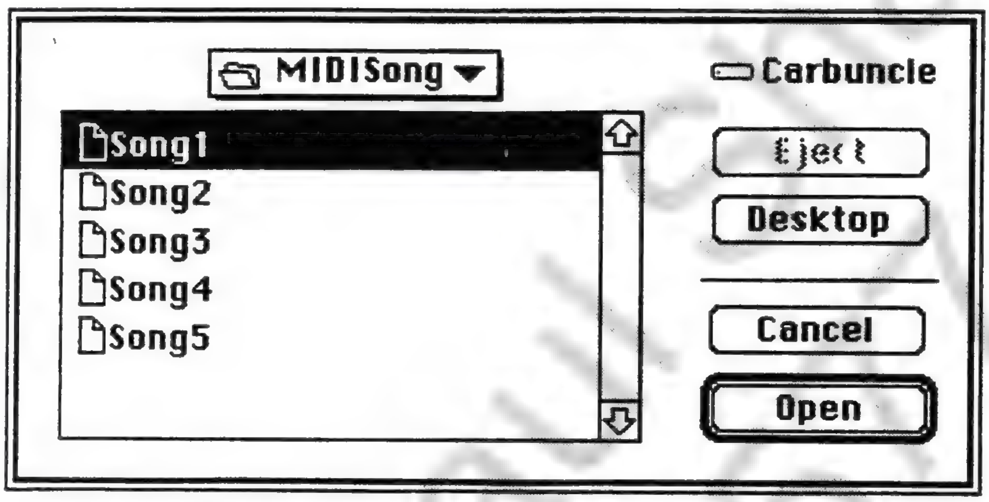
Close
Closes a map that is being edited. The Sound Simulator session will not end.

Save
Saves the map being edited.

Save as...
Saves the map being edited under a different name.

Save Binary File
Saves the currently active file as a binary file.

Collect List Open
Opens the collect list file. Selecting this displays the following dialog box:



The existing collect list files are displayed. A collect list is a compilation of multiple file names. It eliminates the need to select individual files at the beginning of each work session. It is used for converting standard MIDI files and creating sequence banks. All files opened here contain both of the above file types.

Collect List Save
Saves the currently active collect list file.

Collect List Save As...
Saves the currently active collect list file under a different name.

Function Key Open
Opens the saved function key setups into memory.

Function Key Save
Saves the function key setups to a disk file.

Function Key Save As...
Saves the function key setups under a different name. Function keys are stored in a map file. This menu item is used when function key setups are used in a different map file.

System Startup
Starts the sound system with the sound driver and other sound data transferred to the sound board in advance.

Make Current
Activates the currently selected map, thereby switching the map. The sound driver then uses this map as the current map (currently valid map). If auto load is specified in the map file data, file data is automatically transferred to the sound board.

System Down Load
This feature is used when the target board has crashed and must be reset. It transfers the system startup data to the sound board and eliminates the need to restart the Sound Simulator.

In order to download a customized sound driver, save the driver file under the file name `SDDRV.TSK` in the same folder as the Sound Simulator and execute this command.

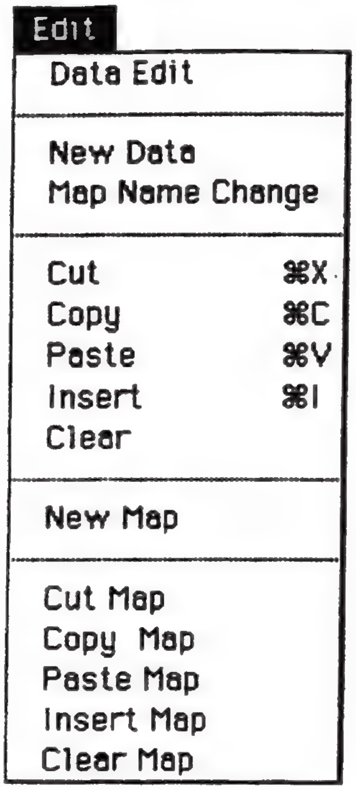
Down Load
Transfers the currently selected file data to the target. Clicking on the download file causes the file to be highlighted in black (reverse type) to show that it has been selected. To select several files at once, hold down the **SHIFT** key while clicking on the desired files. To deselect a file, click on it again to return it to normal display.

Effect Change
Changes DSP programs when there are multiple DSP programs in the current map. Clicking on a DSP program to be switched causes it to be highlighted in black (reverse type) to show that it has been selected. More than one DSP program cannot be selected.

Quit
Ends the Sound Simulator session and closes all open windows.

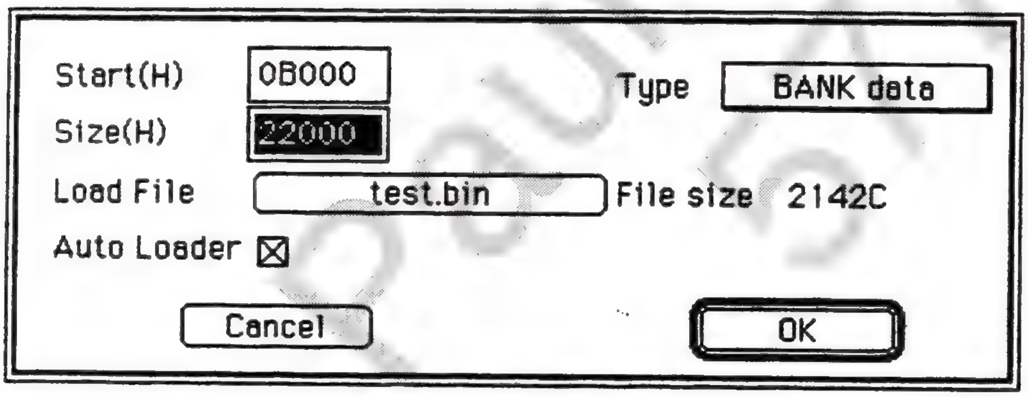
Edit Menu

Selecting this menu displays the following:



Data Edit

Selecting this menu item opens the following dialog box:



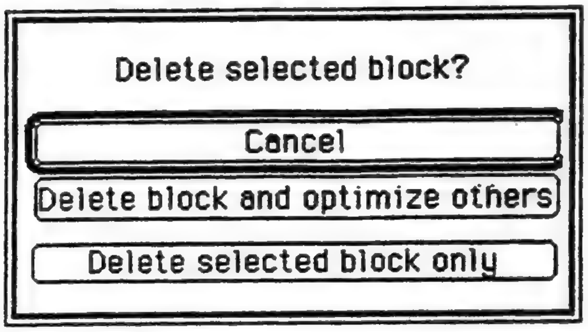
A part of the map data can be edited here. The start address, area size, data type, transfer file, and auto load settings can be set. Specifying the transfer file causes the specified file size to be displayed in bytes to serve as a guide for determining and changing the area size.

Using this menu is the same as double-clicking on the Edit window.

New Data
Selecting this menu item adds new data to the current map. The data type is the default "BANK data" or a data type last selected in the Data Edit dialog box. The default size is set to `00000`.

Map Name Change
Each area map has its own name. This makes it possible to assign easy to understand names when multiple areas exist.

Cut



When selected, a confirmation dialog box appears to prompt the user to either update the address or retain the current address. The cut data is stored in the Clipboard to allow retrieval when **Paste** or **Insert** is selected.

Copy
The selected data is stored in memory and can be retrieved when either **Paste** or **Insert** is selected.

Paste
The most recently stored data in memory is pasted when this is selected.

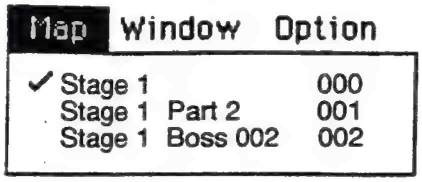
Insert
The most recently stored data is inserted immediately before the selected data. The insertion position is immediately above the selected data, which is highlighted in reverse type. When data is not selected, the inserted data is added at the very end.

Clear
Clears all selected data, with the exception of the data type.

New Map, Cut Map, Copy Map, Paste Map, Insert Map and Clear Map
These functions perform in the same manner as the features above, except maps are edited instead of data.

Map Menu

Clicking on this menu item displays something like the following:

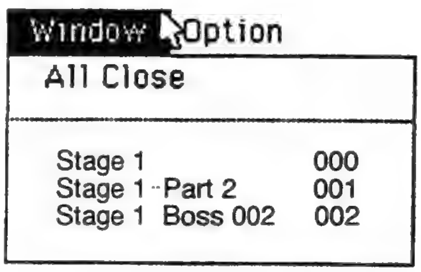


As shown above, the area map to be made current is selected. The area maps switched here are those that are being edited. This operation does not switch the current map. The difference between the Map menu and the Window menu is that the latter only displays other map windows and cannot be used to select other maps.

A maximum of 128 individual map data can be stored with the Sound Simulator.

Window Menu

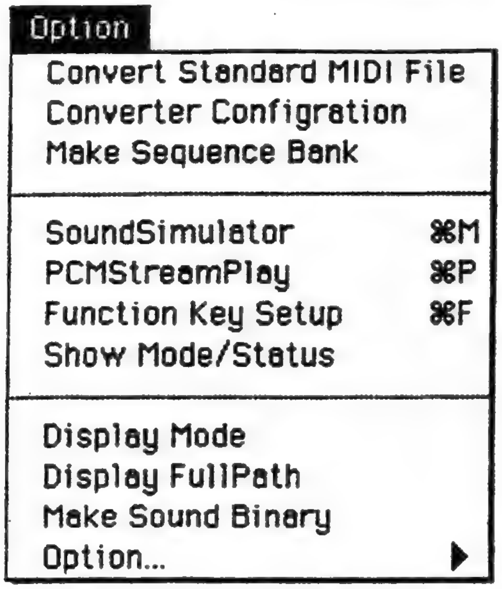
Clicking on this menu item displays the following:



Use this menu to refer to other map data.

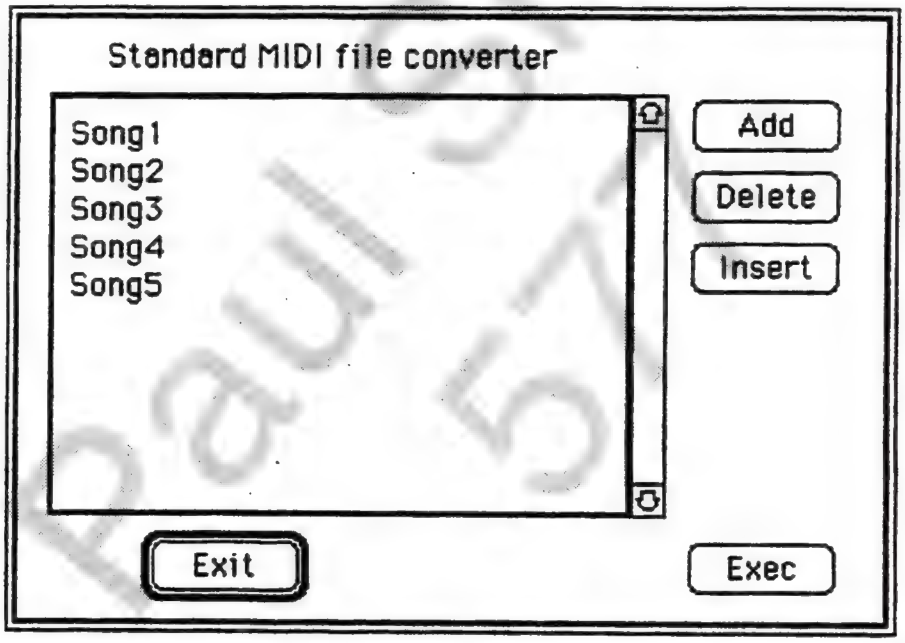
Option Menu

Selecting this menu displays the following:



Standard MIDI File Converter

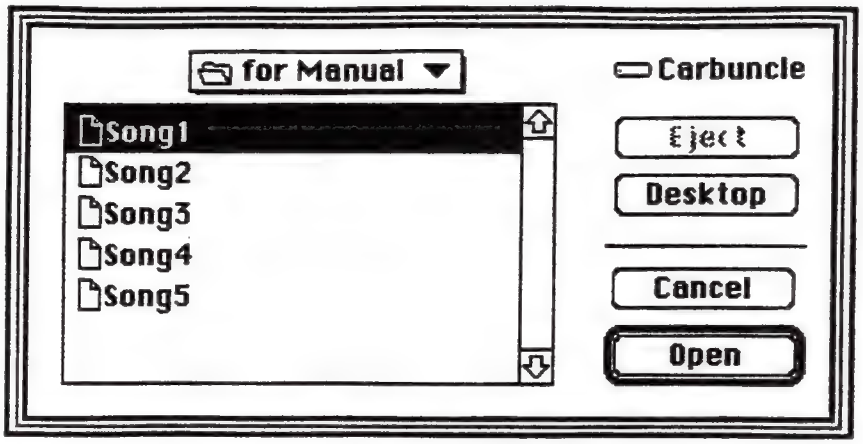
Selecting this menu item from the Option menu opens the following dialog box:



Displayed above are a list of song names selected using the **Add** or **Insert** buttons that are referred to as a "collect list". Nothing is displayed until titles are added or inserted using the buttons. The **Collect List Open** menu item allows the collect list to be read without having to use **Add** or **Insert** to register songs for each Sound Simulator session.

Multiple files can be selected and converted by holding down the **SHIFT** key while clicking on them.

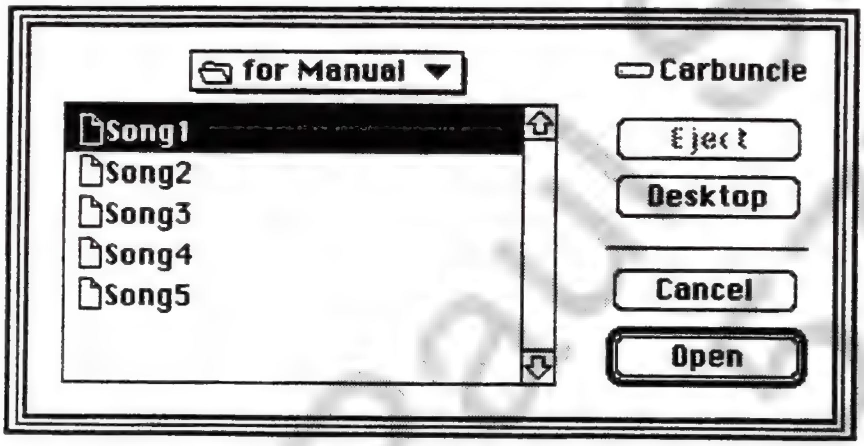
Add
Clicking this button opens the following dialog box:



Song names are added to the collect list here. When a song name is selected, it is added to the end of the list.

Delete
Select a song name from the collect list and click the **Delete** button to remove it from the list.

Insert
Clicking this button opens the following dialog box:

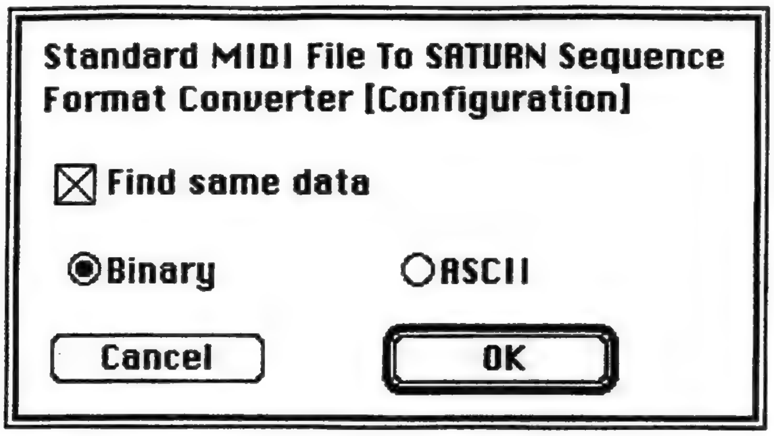


Song names are inserted into the collect list here. Click on the location in the collect list to insert the song name and then select the song. The song name is then inserted directly above the selected location.

Exec
Start up the converter by specifying the song name (a standard MIDI file) to be converted and then click on **Exec**. A compressed SATURN format file is output. The output file format and compression mode can be specified with the **Converter Configuration** menu item from the Option menu. The final output file is output with the specified file name plus the extension .CNV.

Converter Configuration

Select this menu item from the Option menu to display the following dialog box:



An explanation of each check box is given below.

<u>Find Same Data</u>

This analyzes and deletes data that is repeated more than twice in the playback data. This method of compression does not affect playback.

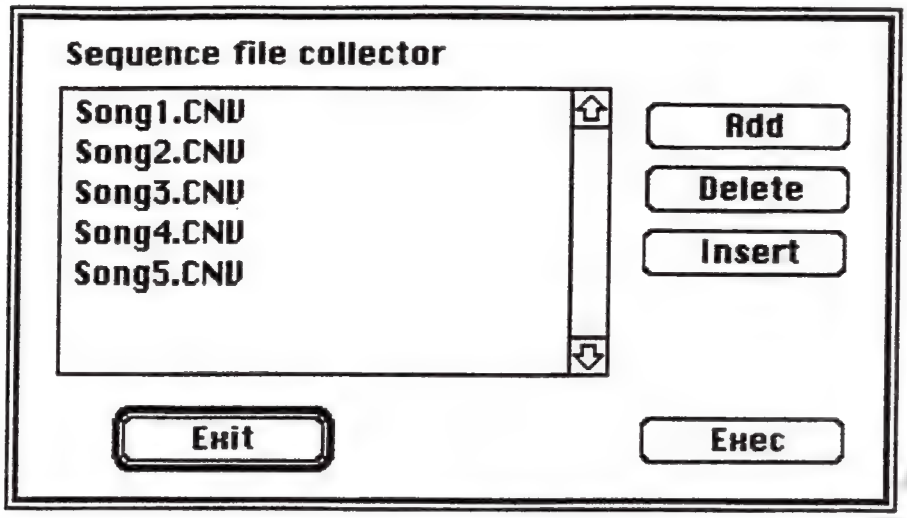
Since this procedure may be time consuming (more song data increases the amount of compression time), this box should only be checked during the final conversion process.

<u>Binary/ASCII</u>

This feature allows the user to select whether the final output file is in ASCII or binary format. If the file is output in ASCII format, the MIDI note data and control changes can be easily checked.

Make Sequence Bank

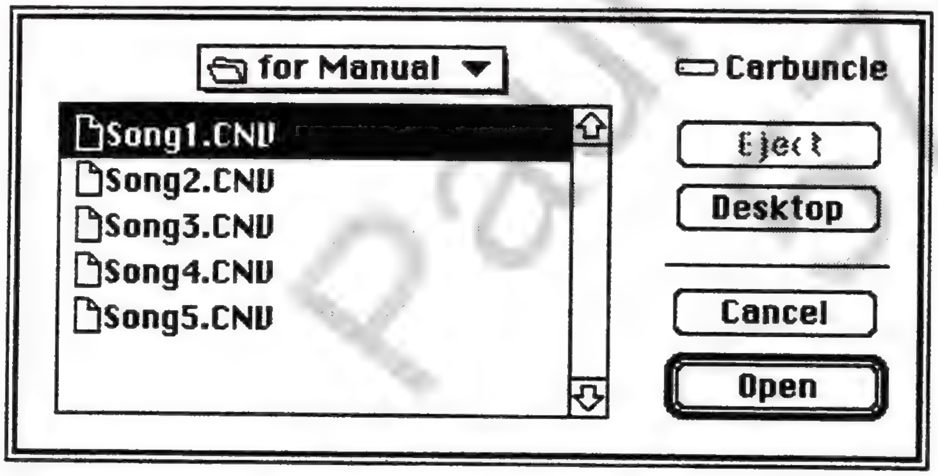
Selecting this menu from the Option menu opens the following dialog box:



A list of compressed file names (called a collect list) converted by the Sound Simulator's standard MIDI file converter is displayed here. Nothing is displayed at first until file names are added or inserted using the **Add** or **Insert** buttons. The **Collect List Open** menu item allows the collect list to be read without having to add or insert a file name at the beginning of each Sound Simulator session.

<u>**Add**</u>

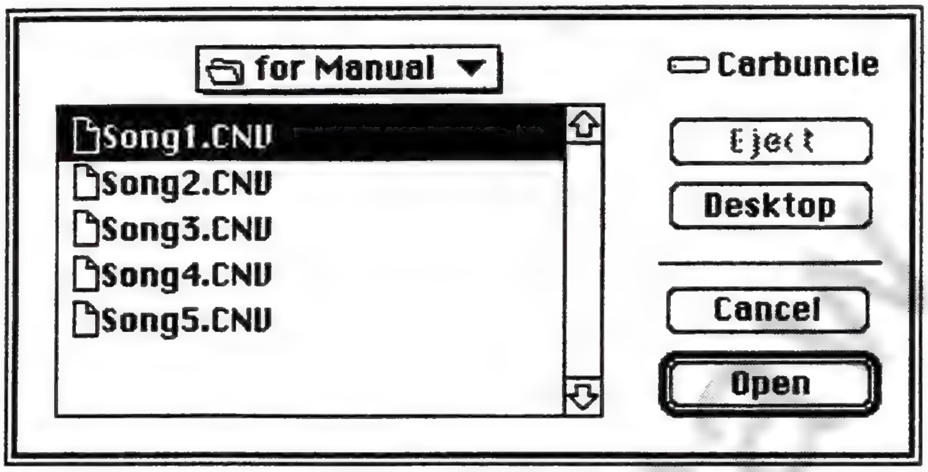
Selecting **Add** opens the following dialog box:



Compressed data file names are added to the collect list here. Selecting the compressed data file name adds that file to the end of the list.

Delete
Select a compressed data file name from the collect list and then click the **Delete** button to delete the file.

Insert
Selecting this displays the following dialog box:

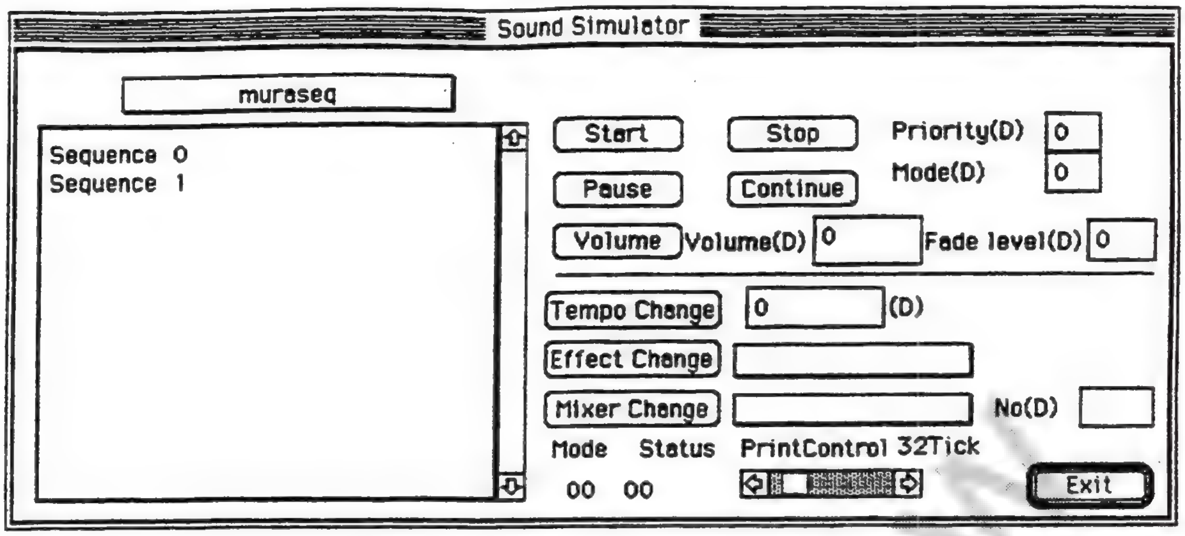


The compressed data file name is inserted in the collect list here. Click on the area in the collect list to insert the file and then select the compressed data file name. The compressed data file name is added to the collect list immediately above the file that was selected.

Exec
Links all the compressed data files specified in the collect list and outputs them as one sequence data bank. The name of the output file can be specified freely.

Sound Simulator

Selecting this menu item displays the following window:



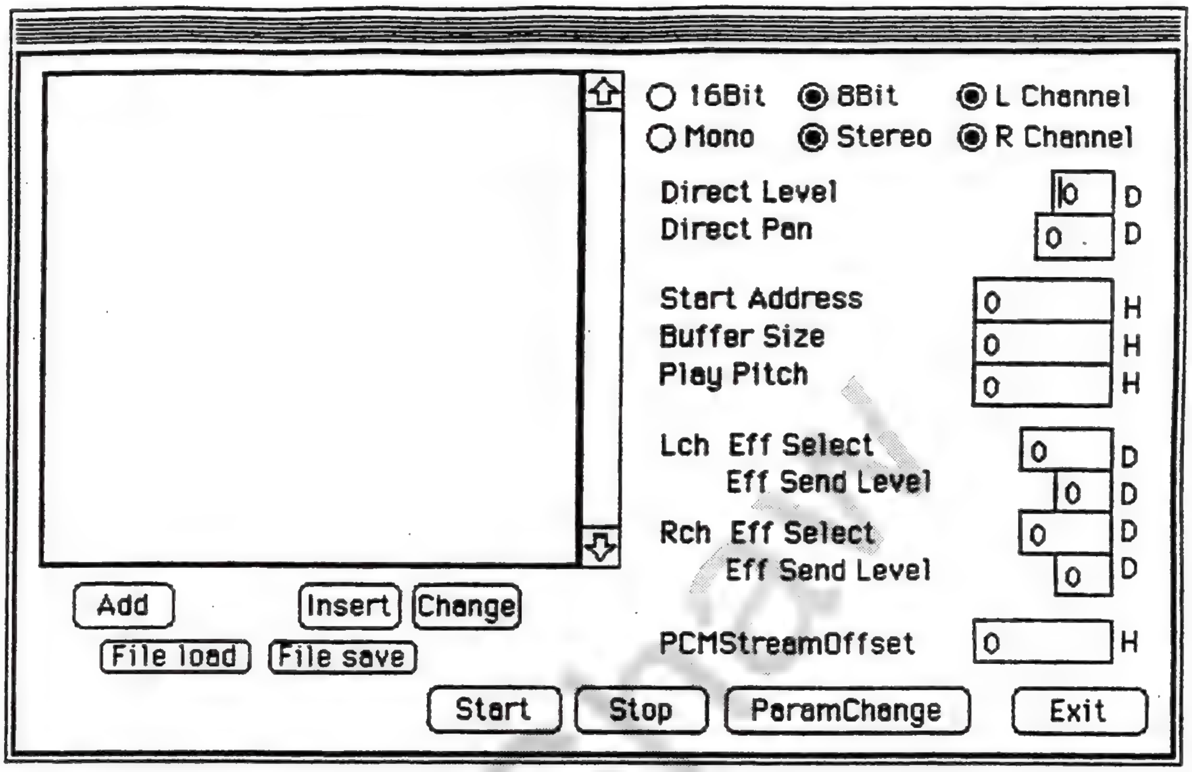
If sequence banks have been transferred from the currently active map, then they can be selected from the pull-down menu above the file list box. Selecting the sequence bank displays a list of the sequence names in the bank similar to the one shown above. Select the sequence desired for playback and use the buttons described below to control playback.

- **Tempo Change:** Changes the tempo of the song.
- **Tempo:** Specifies the value of the tempo to be changed. A plus (+) value means faster, a minus (-) value means slower. The following is an example:
  With a setting range of-32786 to +32767,
  4096 equals 2x (1/2 when negative)
- **Start:** Starts playback.
- **Stop:** Stops playback.
- **Volume:** Sets the sequence volume level and the speed of Fade-in (out).
- **Fade level:** Sets the fade level of Fade-In/Fade-Out. Setting ranges from 0 to 255, where 255 is the slowest.
- **Volume:** Sets the volume from 0—>127, with 127 being the maximum volume. Volume is used in combination with Fade Level.
- **Pause:** Temporarily stops playback.
- **Continue:** Restores playback from pause point.
- **Priority:** Specifies sequence playback priority from 0—>127, with 0 having the highest priority.
- **Mode:** Sets the various modes. See the *SATURN Sound Driver Manual* for more information.
- **Effect Change:** Changes the DSP program. Selection is made from a pull-down menu.
- **Mixer Change:** Changes the mixer. Selection is made from a pull-down menu.
- **No:** Sets the mixer number (0 to 127).
- **Mode:** Displays what task the sound driver is currently performing.
- **Status:** Displays the current status of the sound driver.
- **Print Control:** Sets the interrupt interval for Mode and Status and can be set from 8 to 200 ticks, where one tick is 1/60th of a second.
- **Exit:** Ends the simulation and closes the window.

Note: Refer to the section on *System Interface* for a detailed description of each control command.

PCM Stream Play

Selecting this menu item displays the following window:



Add, Insert

Enters a new file for PCM stream data playback. The file can be either an AIFF or BIN file.

Change

The file is changed without changing Direct Level, Direct Pan and other settings in the file.

Delete

The entered PCM files are deleted along with its settings.

File load, File save

The full path of the entered file and the file settings are saved.

Start

Starts PCM Stream Play according to the settings.

Stop

Stops PCM Stream Play.

Param Change

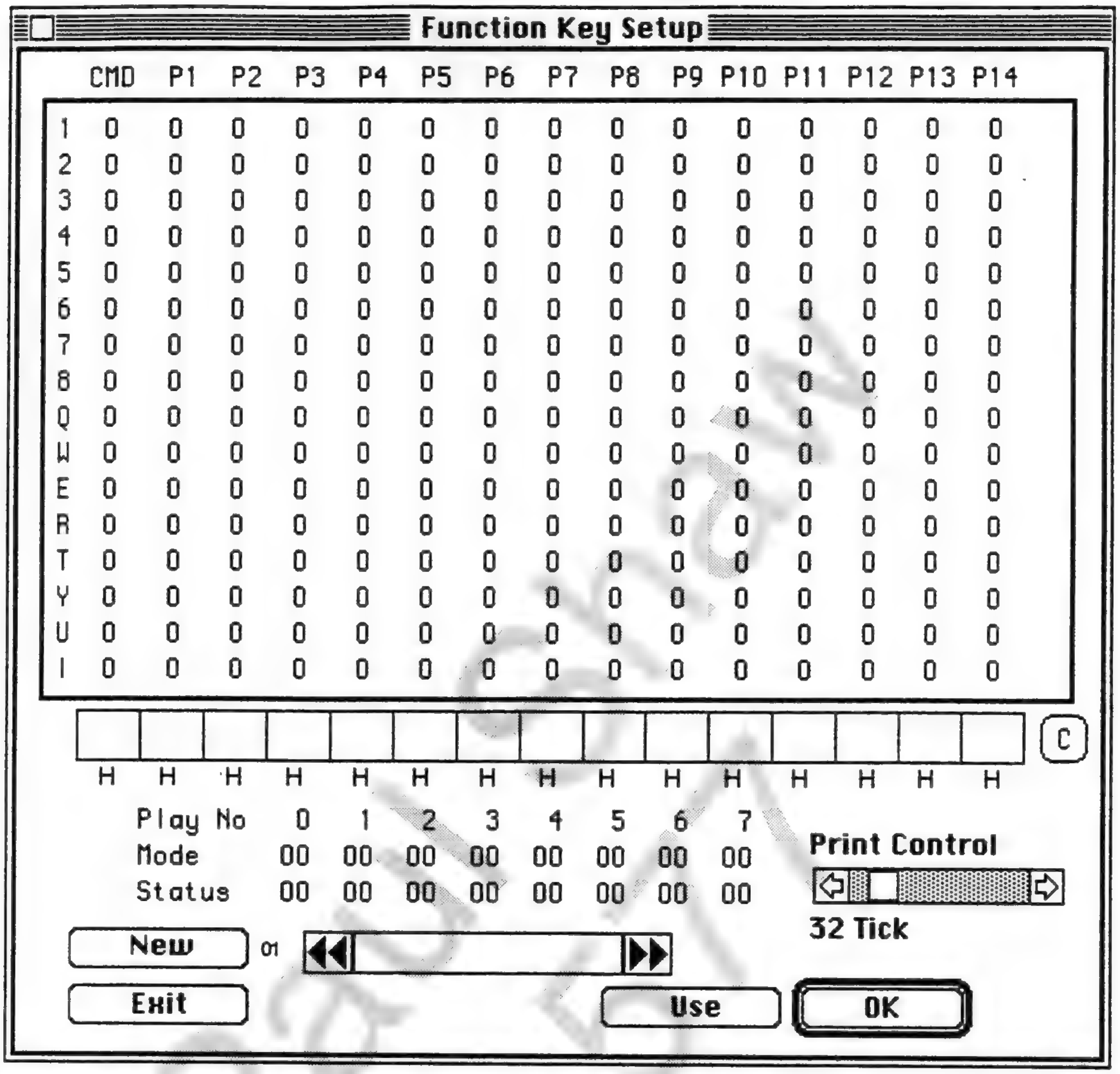
Changes the settings of the PCM data being played back. The parameters for which settings can be changed are Direct Level, Direct Pan, Play Pitch, Eff Select of the left and right channels, and Eff Send level of the left and right channels.

Exit
Ends the PCM Stream Play mode and closes the dialog box.

- **Parameters**
  - **16 bit, 8 bit**
    Selects 16 bits or 8 bits for the PCM playback data.
  - **Mono, Stereo**
    Selects stereo or mono for the PCM playback data. These parameters are not automatically detected even if an AIFF file is loaded.
  - **R Channel, L Channel**
    Enables the specification of the output channel for stereo audio file playback. This setting cannot be made during playback of the file. This parameter is ignored by the **ParamChange** button.
  - **Direct Level**
    Sets the volume during playback from a range of 0 to 8.
  - **Direct Pan**
    Sets the Pan for playback from a range of 0 to 31. This is ignored in stereo playback.
  - **Start Address**
    This is the start address of the PCM playback data. The setting ranges from `0` to `0xffff0`, with the setting at position 1 being ignored.
  - **Buffer Size**
    Sets the buffer for the PCM playback data from a range of `0` to `0xf000` in `0x1000` units. This value represents the number of samples.
  - **Play Pitch**
    Sets the playback pitch from a range of `0` to `0xffff`.
  - **Eff Select**
    Selects the effect for each of the left and right channels from a range of 0 to 15.
  - **Eff Send Level**
    Sets the Effect Send Level from a range of 0 to 7. Effect-related settings for mono audio are set on the right channel.
  - **PCM Stream Offset**
    Sets the playback position in the buffer from a range of `0` to `0xffff`.

Function Key Setup

Selecting this menu item displays the window below. The functions of the sound driver can be assigned to the number keys 1~8 (includes numeric keypad) and the keys Q, W, E, R, T, Y, U and I.



- **Print Control:** Adjusts the interrupt interval for updating Mode and Status. One tick is 1/60th of a second.
- **C:** Clears the contents of the selected function key.
- **New:** Prepares a new map page.
- **<—/—>:** Selects the page for key assignment. Function key tables are identified by their number and name. The name of a table can be set as desired by the text edit box between the arrow keys.
- **Use:** Select the page using <— —> and confirm the active key assignment.
- **OK:** Assigns the input data set in the middle input box to the selected key.
- **Exit:** Closes the window.

Refer to the *SATURN Sound Driver Manual* for more information on the various command codes and parameters.

Print Mode/Status

Selecting this menu item displays the following dialog box which shows the current mode and status of the target:

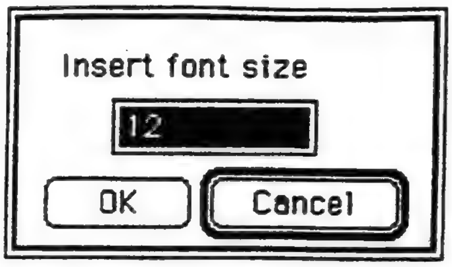
Mode/Status

							32Tick	
Play No	0	1	2	3	4	5	6	7
Mode	00	00	00	00	00	00	00	00
Status	00	00	00	00	00	00	00	00

The update rate of the mode and status data is set by the Print Control parameter setting in the **Sound Simulator** and **Function Key** setup windows. Click on the close window box to close this window.

Display Mode

The dialog box below is displayed when this menu item is selected:



Specifies the font size of the displayed characters on-screen. A larger font size can be selected if the characters are difficult to read.

Print FullPath

The dialog box below is displayed when this menu item is selected:

If a data file or folder has been moved, it will not be read in because the file is not at the last-specified location of the hard disk. Select the file name by clicking on it, then click **Print Full Path**, to display the full path for the folder in which the file is registered.

Make Sound Binary

This command reads in each file according to the current map and outputs a data file equivalent to the data sent to the target. The download files that are converted have the full file path settings. The file size is 479KB from `0xB000` to `0x80000`. Sections where no full file path data exist are padded with `0x00`.

Option

The following submenu is displayed when the Option menu item is selected:

Make CompleteCode

Beginning with the sound driver running on the target board, all data is made into a single file based on the current map. The difference of this feature from the **MakeSoundBinary** menu item is that that data already transferred to the target board are retrieved. This function notifies the user at compile-time as to whether it is enabled or not. If a map exists but the function does not work, it means that the function cannot be enabled.

Send Driver to SBox

Downloads a sound driver to the sound development target box. Since the download address starts from 0, once it has been executed, this menu is disabled regardless of the driver.

To start a customized driver, place the sound driver file with the name `SDDRVS.TSK` in the same folder as the Sound Simulator and then execute this command.

SEGA™

SEGA OF AMERICA, INC.
Consumer Products Division

Sound Editor
User's Manual

Doc. #ST-101-031594

NOTICE

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.

2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without SEGA's permission.

3. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

4. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

5. No license is granted by implication or otherwise under any patents or other rights of any third party or SEGA Enterprises, Ltd.

6. This document is confidential. By accepting this document you acknowledge that you are bounded by the terms set forth in the non-disclosure and confidentiality agreement signed separately and/in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

(4/20/94 - 001)

READER CORRECTION/COMMENT SHEET

Keep us updated!

If you should come across any incorrect or outdated information while reading through the attached document, or come up with any questions or comments, please let us know so that we can make the required changes in subsequent revisions. Simply fill out all information below and return this form to the Developer Technical Support Manager at the address below. Please make more copies of this form if more space is needed. Thank you.

General Information:

Your Name ______________________ **Phone** ____________

Document number ST-101-031594 **Date** ____________

Document name Sound Editor User's Manual

Corrections:

Chpt.	pg. #	Correction

Questions/comments: ______________________

____

____

____

Where to send your corrections:

Fax: (415) 802-3963
Attn: Manager,
Developer Technical Support

Mail: SEGA OF AMERICA
Attn: Manager,
Developer Technical Support
275 Shoreline Dr. Ste 500
Redwood City, CA 94065

TABLE OF CONTENTS

1.0 Summary

This program enables the display and editing of tone parameters using an SCSP connected to the SCSI port of Apple Macintosh computers.

This program handles the following seven operations.

1) File management
   File management tasks including input/output and saving.

2) Editing
   Data editing operations including undo, cut and paste, and insert.

3) Number processing
   Increases voice, layer, mixer, velocity, PEG, and PLFO values.

4) Window processing
   Open/close of mixer, velocity, PEG, PLFO, FM, and monitor windows.

5) FM processing
   Updates all layers with the same start note and end note as the selected layer.

6) Preferences processing
   Displays the SCSI ID, SCSPBIN format downloading, and voice data display.

7) SCSI processing
   Performs SCSI operations on the Macintosh during I/O of tone data to the Macintosh from the SCSP through the SCSI port.

2.0 Terminology

SCSP format
The file containing all data required for display on the Macintosh.

SCSPBIN format
The file containing the minimum data required to play a sound on the SCSP. The data in this file is actually downloaded to the SCSP.

Voice
The tone. The number associated with this window is the same as the MIDI program change number.

Layer
The sound data. Plural layers are combined to form one voice. In the case of FM, each module and each carrier is treated as a layer.

Wave
One type of sound data; used in reference to the PCM data of an actual sound.

3.0 Tutorial

The minimum operating procedure required to play a sound is described in this tutorial.

- Start the tone editor.
  The tone editor first checks for the presence of the SCSP board in one of the Macintosh slots; if the board is not found, the program quits.

- Select New from the File menu.
  A dialog box requesting the number of voices to be set will be displayed (see Appendix). Here the number of voices is input. First, input 1.

- The voice window will be displayed. (see Appendix).
  Note that only one line is displayed at this time because the specified number of voices was 1. Also note that the number of voices can be changed later.

The following parameters can be set from the voice window: voicename,VendRange, portament time, volume bias, and play mode. All settings other than the play mode can be set by clicking the values; a pop-up menu is displayed for the play mode.

- Double-click the voice name to display a dialog box enabling the voice name to be changed.
  First, input the name. Double-click the "untitled0." The name input dialog box is displayed. Input a name. Any name can be input

  Managing the voices later may be simplified by assigning a name appropriate to the tone you will create.

- Click the OK button to accept the name change and close the dialog box.
  The new name just entered should be displayed in the voice name area. The next step is to create the tone.

- Double-click the number before the voice name to display the dialog box for setting the number of layers. (see Appendix).

- Enter the number of layers as you did for the number of voices. Enter "1" now.

- The layer window will be displayed (see Appendix).
  Again, only one line will be displayed, but this shouldn't cause any concern since it can be changed later.

  The layer window functions as the mixer, making it possible to adjust the output and set the channel and level input to the effector. The minimum operation required here to output a sound is to set the output to maximum and decide the range of the note of the MIDI device playing this sound.

- Click the END number.
  The insertion mark will be displayed. Enter 127.

- Move the DirectLevel slide bar all the way to the right.
  The number beside the slide bar should be 7.

- Enter a name for the layer. The procedure is the same as entering the voice name.

This completes the output settings. The parameters must now be set.

- To set the parameters, double-click the number beside the layer name.
  The slot window will be displayed (see Appendix).

  The parameters are divided into individual blocks, each with its own special window. Again, only the minimum parameters required to play a sound will be set at this time.

- To display the wave data (AIFF data) input window, double-click the blank rectangle in the middle (see Appendix).

- Select the AIFF file and click OK.
  With the slot window still displayed, the AIFF filename will appear in the previously empty rectangle.

- If there is a loop in the selected wave data, double-click the slot window inverter (small triangle with a circle around it).
  The invert window will be displayed (see Appendix).

- Select the loop mode from the loop selection pop-up menu.

Do the following to play a sound:

- Double-click the picture of a mountain in the slot window.
  The EG window will be displayed (see Appendix.).

- Set the AR and RR sliders all the way to the right (31 will be displayed).

- For the final setting, double-click the TOTAL LEVEL in the slot window.
  The total level window will be displayed (see Appendix).
  The velocity is selected from a pop-up menu, but is still not set. To set the velocity, first close the total level window.

- Select Velocity from the Window menu (see Appendix).
  The velocity window will be displayed (see Appendix).
  By now you should be familiar with how to use this window: double-click the name to enter the name, and double-click the number beside the name to display the Velocity Edit window (see Appendix).

- Double-click the name and enter a meaningful name.
- Double-click number 0.
- Move the L0, L1, L2, and L3 sliders all the way to the right.
- Since the screen will become hard to view, close the Velocity Edit and velocity windows.
- Now display the total level window again. The velocity pop-up menu should now contain the name just assigned.

  When creating additionally No. 1, No. 2 via the velocity window, then select from the pop-up menu. A sound can now be played. Try playing the MIDI device connected to the SCSP board with program change 0.

5.0 Function Descriptions

5.1 Menu Bar

The menu bar contains the following menus (see Appendix).
(1) Apple menu
(2) File menu
(3) Edit menu
(4) Number menu
(5) Window menu
(6) FM menu
(7) Preference menu

5.1-1 Apple Menu

This is similar to the standard Apple menu.

5.1-2 File Menu

The File menu contains the following commands (see Appendix).

1. New
2. Open ...
3. Save
4. Save As
5. Close
6. Quit

5.1-3 Edit Menu

The Edit menu contains the following commands (see Appendix).

1. Cut
2. Copy
3. Paste
4. Insert

5.1-4 Number Menu

The Number menu contains the following commands (see Appendix).

1. Voice
2. Layer
3. Mixer
4. Velocity
5. PEG
6. PLFO

5.1-5 Window Menu

The Window menu contains the following commands (see Appendix).

1. Mixer
2. Velocity
3. PEG
4. PLFO
5. FM
6. MONITOR

5.1-6 FM Menu

The FM menu contains the following commands (see Appendix).

1. UPDATE

5.1-7 Preference Menu

The Preference menu contains the following commands (see Appendix).

1. SCSI info
2. Download
3. Voice Info

5.2 Master Volume Window

The master volume window is always displayed. The master volume can be changed from this window, and the master volume setting can be set at the same time (see).

5.3 File Menu Process

5.3-1 New

Opens the number of voices window to set the number of voices. A new tone library is opened when OK is clicked.

5.3-2 Open

Loads a tone format file. Only SCSP format files can be loaded. Double-click the filename to open the voice window and the sound mixer window (see Appendix).

5.3-2-1 Voice Window

Changes the voice monitor and parameters. The following parameters can be set from this window (see Appendix).

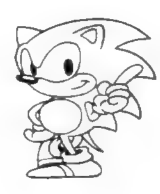
1) Tone name
   Displays the tone name. Double-click the name to open the name change window. The name can be changed from the window. (see Appendix). Double-click the number beside the name to open the layer window.

2) VendRange
   Sets the adjustable range of the VendRange value.
   Values from 0 ~ 12 can be set. This value expresses the change in pitch as a MIDI note when the pitch Vend wheel is moved to the maximum setting.

3) Portament
   Sets the portament time. Values in the range 0 ~ 127 are permitted.

4) VOLBias
   Sets the volume bias. Values in the range -128 ~ 127 are permitted.

5) MODE
   Displays the play mode.
   POLY, MONO, LEGATO, PORTA, L&P mode selections can be made from the pop-up menu (see Appendix).
   - PORTA is an abbreviation for PORTAMENTO. L&P is an abbreviation for LEGATO and PORTAMENTO.



5.3-2-2 Layer Window

This window is opened by double-clicking the voice name. Layer parameters can be changed from this window. The parameters set from this window are as follows (see Appendix):

1) FM connection carrier check box
   Base slot used when ON (selected) (if plural boards are installed).
   This is used in the FM connection window. For further details, see the FM connection window section.

2) Layer name
   Displays the tone name. Double-click the name to change the name; a dialog box will be displayed (see Appendix). Double-click the number to the left of the name to open the slot window.

3) START
   Sets the start note.
   Values in the range 0 ~ 127 are permitted.
   Sets which MIDI note is played in this layer. When other MIDI notes come between the START and END notes, the sound from this layer is played.

4) END
   Sets the end note.
   Values in the range 0 ~ 127 are permitted.
   Sets which MIDI note is played in this layer. When other MIDI notes come between the START and END notes, the sound from this layer is played.

5) Direct level
   Sets a direct send. Values in the range 0 ~ 7 are permitted; the maximum output level is reached at 7.

6) Effect send
   Sets an effect send. Values in the range 0 ~ 7 are permitted; the maximum output level is reached at 7.

7) Effect select
   Sets the effect input channel.
   Values in the range 0 ~ 15 are permitted.

8) Direct PAN
Sets PAN.
Values in the range R16 ~ C ~ L16 are permitted.

5.3-2-3 Mixer Window

Open this window by selecting Mixer from the Window menu (see Appendix); enables the mixer name to be changed.

5.3-2-4 Mixer Edit Window

(See Appendix)

This window is opened by double-clicking on the left side the mixer window name. Change mixer parameters. The following parameters can be set from this window (see Appendix).

1) Effect level
Sets an effect send/return. Values in the range 0 ~ 7 are permitted; the maximum output level is reached at 7.

2) Effect PAN
Sets an effect PAN.
Values in the range R16 ~ C ~ L16 are permitted.

5.3-2-5 Monitor Window

Open this window by selecting MONITOR from the Window menu (see Appendix). When this window is displayed, the MIDI device can be monitored in real-time through the SCSI port. MIDI channel program changes (SCSP voice name), notes, and velocity are displayed.

1) MIDI
   MIDI channel
   Displayed as a fixed value in the range 0 ~ 31.

2) VOICE
   Displays the SCSP voice name corresponding to the program change of the MIDI channel received through the SCSI port.

3) NOTE
   Displays the NOTE received through the SCSI port.

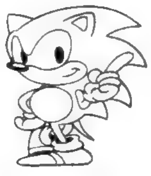
4) VELO
   Displays the velocity number received through the SCSI port.

5.3-2-6 FM Connection Window

Open this window by selecting FM from the Window menu (see Appendix).
The FM connections of the currently selected tones are displayed.
The following settings are required to use the layer as FM.

- Set the modulator and carrier to the same START and END values.
- Add a carrier check button to the carrier.
- Set the FM connection data in the slot window.

When the FM connection window is opened or the FM connection window is updated from the FM menu after the above settings are made, the FM algorithm will be displayed in the FM connection window. The number used at this time is the number of each layer in the layer window.



5.3-2-7 Slot Window

This window is opened by double-clicking the number in front of the layer name of the layer window. Multiple slot windows can be displayed simultaneously. The parameter window applies to all slot windows, and only one parameter window can be displayed.

SCSP register parameters can be edited, and the parameter window can be opened (see Appendix). Double-click the corresponding icons to open PEG, PLFO, ALFO, WAVE.AIFF, inverter blocks, EG, SLOTOUT, MODULE1, MODULE2, and MDL.

- The switches toggle on/off with each click.
- WAVE.AIFF displays the AIFF filename used in that slot.
- MODULE 1, 2 display the layer name, and are disabled when the LEVEL (MDL) is 0.

If the slot window is moved while holding the shift key down, the parameter window opened from that slot window also moves.

5.3-2-8 Parameter (PEG) Window

This window is opened by double-clicking the PEG block in the slot window (see Appendix). This window enables selection of the parameters from the PEG and PLFO windows using a pop-up menu.

1) PEG (PEON)
   Sets PEG on/off.

2) MODULATION (MWE)
   Turns PEG adjustment using the MODULATION wheel on/off.

3) PEG pop-up menu
   Enables selection of the parameters created in the PEG window.

4) PLFO (PLON)
   Sets PLFO on/off.

5) MODULATION (MWL)
   Turns PLFO adjustment using the MODULATION wheel on/off.

6) PLFO pop-up menu
   Enables selection of the parameters created in the PLFO window.

5.3-2-8 LFO Window

This window is opened by double-clicking the PLFO or ALFO block in the slot window. Parameters relating to hardware LFO are edited from the LFO window. The following parameters are set from this window (see Appendix).

1) MODULATION
   Turns the hardware LFO bit on/off.

2) LFO RESET (LFORE)
   Resets LFO.

3) LFO FREQUENCY (LFOF)
   Sets the LFO oscillation frequency.
   Values in the range 0 ~ 31 are permitted.

4) LFO WAVE (PLFOWS)
   Sets the shape of the PLFO waveform.
   SAW, SQUARE, TRIANGLE, and NOISE can be set.

5) LFO DEPTH (PLFOS)
   Sets the degree of effect with respect to LFO pitch.
   Values in the range 0 ~ 7 are permitted.

6) LFO WAVE (ALFOWS)
   Sets the shape of the ALFO wave.
   SAW, SQUARE, TRIANGLE, and NOISE can be selected.

7) LFO DEPTH (ALFOS)
   Sets the degree of effect mixing with respect to LFO EG.
   Values in the range 0 ~ 7 are permitted.

5.3-2-9 WAVE.AIFF Window

This window is opened by double-clicking the WAVE.AIFF block (normally white rectangular) in the slot window (see Appendix). This window displays the details of the wave to be selected. When the OK button in this window is clicked, the loop start address (LSA) of the selected AIFF file, the loop end address (LEA) of the sound data, and the parameters set with the radio buttons are written in the SCSP format.

The following parameters can be set from this window.

1) NOISE
   Uses internally generated data (noise) as the sound input data.

2) ALL "0"
   Uses internally generated data (ALL "0") as the sound input data.

3) WAVE
   The layer dialog is displayed when OK is pressed.

4) Listen
   Plays the selected file through the Macintosh speaker.

5) OK
   Captures the waveform.

5.3-2-10 Inverter Window

This window is opened by double-clicking the inverter block in the slot window (see Appendix). This window enables bit inversion of the sound input data, and editing the loop type, layer sample note, and FINETUNE parameters.

The following parameters are set from this window.

1) SIGNFRAGINVERT (SPCTL1)
   Inverts the sign bit of the sound input data.

2) DATAINVERT (SPCTL0)
   Inverts all bits other than the sign bit of the sound input data.

3) LOOP SELECT
   Selects the loop type from the following options:
   - FORWARD
   - REVERSE
   - ALTERNATE
   - OFF

4) BASE NOTE
   Sets the base note. Values in the range 0 ~ 127 are permitted.
   When an AIFF file is loaded, the BASENOTE value in the AIFF file is automatically loaded to set this parameter.

5) FINETUNE
   Sets the fine tune parameter.
   Values in the range 0 ~ 127 are permitted.
   When an AIFF file is loaded, the FINETUNE value in the AIFF file is automatically loaded to set this parameter.

6) SAOFSET
   Sets the SA offset.
   Values in the range 0 ~ 65535 are permitted.
   This parameter is used when setting the wave start in the middle of the wave form data. Used primarily for sine waveform data for FM. For further information, see the SCSP hardware manual.

5.3-2-11 EG Window

This window is opened by double-clicking the EG block (where the picture of a mountain is displayed) in the slot window (see Appendix). This window enables editing EG-related parameters. Parameters can also be changed by dragging EG.

The following parameters can be set from this window.

1) SORHOLD (EGHOLD)
   - ON
     Holds an attack value of "000."
   - OFF
     The attack value varies with the attack rate (AR).

2) ATTACK RATE (AR)
   Sets the change level for EG in the attack state.
   Values in the range 0 ~ 31 are permitted.

3) DECAY RATE1 (D1R)
   Sets the change level for in EG in the decay 1 state.
   Values in the range 0 ~ 31 are permitted.

4) DECAY RATE2 (D2R)
   Sets the change level for EG in the decay 2 state.
   Values in the range 0 ~ 31 are permitted.

5) DECAY LEVEL (DL)
   Sets the attenuation level for EG transition from decay 1 to decay 2.
   Values in the range 0 ~ 31 are permitted.

6) RELEASE RATE (RR)
   Sets the change level for EG in the release state.
   Values in the range 0 ~ 31 are permitted.

7) KEY RATE SCALE (KRS)
   Sets the amount of EG key rate scaling.
   Values in the range 0 ~ 15 are permitted.

8) LPSLNK
   If the address of the loaded sound slot input data exceeds the loop start address, EG changes to decay 1, even if EG is not "000."

5.3-2-13 TOTAL LEVEL Window

This window is opened by double-clicking the TOTAL LEVEL block in the slot window (see Appendix). This window enables editing the volume and pitch parameter.

The following parameters can be set from this window.

1) TOTAL LEVEL (LFORE)
   Sets the attenuation level applied to the EG value.
   Values in the range 0 ~ 127 are permitted.

2) FNSX
   The F number extension bit; uses the DSP modulation signal for the sound data.

3) SDIR
   Outputs the sound data without multiplying EG, TL (volume), and ALFO.

4) STINH
   Prohibits writing the slot output to the data buffer.

5) Velocity
   Selects the velocity.

5.3-2-14 MODULE1 Window

This window is opened by double-clicking the MODULE1 block in the slot window (see Appendix). This window enables setting the source used as modulation input X. Modulation X of the selected layer is set to the selected generation.

The following parameters can be set from this window.

1) LAYER pop-up menu
   Displays the layer name.

2) Generation
   - 0
     Latest data
   - 1
     Data from the last generation

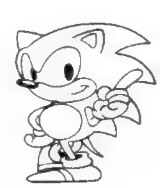
5.3-2-15 MODULE 2 Window

This window is opened by double-clicking the MODULE2 block in the slot window (see Appendix). This window enables setting the source used as modulation input Y. Modulation Y of the selected layer is set to the selected generation.

The following parameters can be set from this window.

1) LAYER pop-up menu
   Displays the layer name.

2) Generation
   - 0
     Latest data
   - 1
     Data from the last generation



5.3-2-16 MDL window

This window is opened by double-clicking the MDL block in the slot window (see Appendix).

The following parameters can be set from this window.

1) LEVEL (MDL)
   Sets the effect (degree of modulation) on modulation of the modulation input source. Values in the range 0 ~ 15 are permitted.

5.3-3 Save

Saves the SCSP file currently in use (see Appendix).

5.3-4 Save As

Saves the SCSP file currently in use under a different filename (permitting the user to change the filename).

The following buttons are used in this window (see Appendix).

1) SCSP
   Saves the file in the SCSP format.

2) SCSPBIN
   Saves the file in the SCSPBIN format.

3) Save
   Displays the Replace dialog, and saves the file if the Replace button is pressed (see Appendix).

5.3-5 Close

Closes the file currently in use. An alert is displayed if the user attempts the close the file without saving it after changes have been made. Windows below the active window are also closed automatically.

When the slot window is closed, all parameter windows are also closed. When the layer window is closed, the slot window is also closed. When the voice window is closed, an alert is displayed if changes had been made.

The following buttons are used in this window (see Appendix).

1) OK
   Closes the window.

2) Save
   Saves the file.

3) Cancel
   Returns to the edit window.

5.3-6 Quit

Terminates the tone editor and returns to the Finder. If any windows are open, an alert is displayed (see Appendix).

5.4 Edit Menu Process

5.4-1 Cut

When in the layer window, cuts the layer from the format and saves to the clipboard. When in the voice window, cuts the layer associated with the voice chunk, and saves the voice chunk data and the associated layer chunk data to the clipboard. PEG, Velocity, and PLFO are cut from the format and saved to the clipboard.

5.4-2 Copy

When in the layer window, copies the layer chunk data to the clipboard. When in the voice window, copies the voice chunk data and the associated layer chunk data to the clipboard. Also copies from PEG, Velocity, and PLFO data to the clipboard.

5.4-3 Paste

Pastes the data from the clipboard. Can only be pasted per window with the same clipboard data. Paste operations are <u>overwrite only</u>.

5.4-4 Insert

Adds one more voice, layer, or mixer.

5.5 Number Menu Process

Number menu process is used to increase the corresponding value.

5.5-1 Voice

Sets the number of voices (see Appendix). It is not possible to set a value less than the current number of voices.

5.5-2 Layer

Sets the number of layers (see Appendix). It is not possible to set a value less than the current number of layers.

5.5-3 Mixer

Sets the number of mixers (see Appendix). It is not possible to set a value less than the current number of mixers.

5.5-4 Velocity

Sets the velocity number (see Appendix). It is not possible to set a value less than the current velocity setting.

5.5-5 PEG

Sets the PEG number (see Appendix). It is not possible to set a value less than the current PEG setting.

5.5-6 PLFO

Sets the PLFO number (see Appendix). It is not possible to set a value less than the current PLFO setting.

5.6 Window Process

Closes the corresponding open windows and opens them if closed.

5.6-1 Mixer

Opens/closes the mixer window.

5.6-1-1 Mixer Window

Enables changing the mixer name (see Appendix).

5.6-1-2 Mixer Edit Window

This window is opened by double-clicking on the left side the mixer window name (see Appendix). Mixer parameters can be changed from this window.

The following parameters can be set from this window.

1) Effect level
   Sets an effect send/return.
   Values in the range 0 ~ 7 are permitted.

2) Effect PAN
   Sets output PAN to which an effect is applied.
   Values in the range R16 ~ C ~ L16 are permitted.

5.6-2 Velocity

Opens/closes the velocity window.

5.6-2-1 Velocity Window

Enables changing the velocity name (see Appendix).

5.6-2-2 Velocity Edit Window

This window is opened by double-clicking on the left side the velocity window name (see Appendix).

The following parameters can be set from this window. The effect of parameter changes can be viewed on the bottom graph.

1) VELOCITY 0
   Sets the velocity 0 (horizontal axis).
   Values in the range 0 ~ 127 are permitted.

2) VELOCITY LEVEL 0
   Sets the level 0 (vertical axis) point of the velocity curve.
   Values in the range 0 ~ 127 are permitted.

3) VELOCITY 1
   Sets the velocity 1 (horizontal axis) parameter.
   Values in the range 0 ~ 127 are permitted.

4) VELOCITY LEVEL 1
   Sets the level 1 (vertical axis) point of the velocity curve.
   Values in the range 0 ~ 127 are permitted.

5) VELOCITY 2
   Sets the velocity 2 (horizontal axis) parameter.
   Values in the range 0 ~ 127 are permitted.

6) VELOCITY LEVEL 2
   Sets the level 2 (vertical axis) point of the velocity curve.
   Values in the range 0 ~ 127 are permitted.

7) VELOCITY LEVEL 3
   Sets the level 3 (vertical axis) point of the velocity curve.
   Values in the range 0 ~ 127 are permitted.

5.6-3 PEG

Opens/closes the PEG window.

5.6-3-1 PEG Window

Enables changing the PEG name (see Appendix).

5.6-3-2 PEG Edit Window

This window is opened by double-clicking on the left side the PEG window name (see Appendix). Through parameters set in this window, the following data could be changed. Parameter changes are immediately reflected in the graph.

1) DELAY TIME (msec)
   Sets the delay time to PEG start.

2) OFFSET LEVEL (cent)
   Sets the offset level from the key-on note to PEG start.

3) ATTACK LEVEL (cent)
   Value of the attack level

4) ATTACK TIME (msec)
   The time to PEG attack

5) DECAY LEVEL (cent)
   Value of the decay level

6) DECAY TIME (msec)
   Time to PEG decay

7) SUSTAIN LEVEL (cent)
   Value of the sustain level

8) SUSTAIN TIME (msec)
   Time to reach the sustain level

9) RELEASE LEVEL (cent)
   Value of the release level

10) RELEASE TIME (msec)
    Time to reach the release level

5.6-4 PLFO

Opens/closes the PLFO window.

5.6-4-1 PLFO Window

Enables changing the PEG name (see Appendix).

5.6-4-2 PLFO Edit Window

This window is opened by double-clicking on the left side the PLFO window name (see Appendix).

The following data is changed by setting the parameters in this window. Parameter changes are immediately reflected in the graph.

1) PLFO DELAY TIME (msec)
   Sets the PLFO delay time.

2) PLFO DEPTH LEVEL (Hz)
   This is the variation range for PLF waveform.

3) PLFO FREQ TIME (cent)
   The time from the 0 point until the PLFO wave reaches the depth level set in 2) above.

4) PLFO FADE TIME (msec)
   The time required for a fade-in to complete

5.6-5 FM

Opens/closes the FM window.
This window displays the FM connections of the currently selected tone (see Appendix).

To use layers as FM, the following settings are required.

- Set the modulator and carrier to the same START and END values.
- Add a carrier check button to the carrier.
- Set the FM connection data in the slot window.

When the FM connection window is opened or the FM connection window is updated from the FM menu after the above settings are made, the FM algorithm will be displayed in the FM connection window. The number used at this time is the number of each layer in the layer window.

5.6-6 Monitor

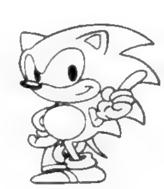
Opens/closes the MONITOR window. When the MONITOR window is displayed, the MIDI device can be monitored in real-time through the SCSI port.
MIDI channel program changes (SCSP voice name), notes, and velocity are displayed.

1) MIDI
   MIDI channel
   Displayed as a fixed value in the range 0 ~ 31.

2) VOICE
   Displays the SCSP voice name corresponding to the program change of the MIDI channel received through the SCSI port.

3) NOTE
   Displays the NOTE received through the SCSI port.

4) VELO
   Displays the velocity number received through the SCSI port.



5-7 FM Process

5.7-1 UPDATE

Layers with the same start and end notes as the selected layer are updated when a layer is selected in the layer window and UPDATE is then chosen.

5.8 Preference Process

5.8-1 SCSI address

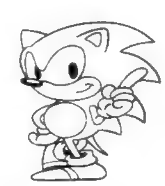
Displays the current SCSI ID (see Appendix).

5.8-2 Download

Downloads the SCSPBIN format tone data to the SCSP board.

5.8-3 Voice Info

Displays the total number of bytes in the tone data currently being edited (see Appendix).



5.9 Error Handling

Processing is stopped and an error dialog box is displayed when an error occurs (see Appendix). This dialog box displays errors during the communication with SCSP as well as other applications errors contents.

5.10 File Descriptions

The following files are used with this application.

1) SCSP data files
   A Macintosh file containing detailed information.

2) SCSPBIN data files
   Contains the data loaded into 68000 memory, and the header data required to create a Macintosh file.

3) Wave Edit data files
   AIFF format files created with the Save operation

4) Alchemy files
   AIFF format files created by Alchemy.

5) Sound designer files
   SD2 format files created by the sound designer

5.11 Other

SEGA

SEGA OF AMERICA, INC.
Consumer Products Division

SOUND TOOL GUIDE

Inside:

NOTICE

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.

2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without SEGA's permission.

3. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

4. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

5. No license is granted by implication or otherwise under any patents or other rights of any third party or SEGA Enterprises, Ltd.

6. This document is confidential. By accepting this document you acknowledge that you are bounded by the terms set forth in the non-disclosure and confidentiality agreement signed separately and/in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

(4/20/94 - 001)

SOUND TOOL GUIDE CONTENTS

1. SCSP Waveform Editor Technical Specifications (ST-67)

2. Technical Specifications for the Sound Editing Tool (ST-68)

3. SCSP/DSP Effect Module Specifications (ST-69)

4. SCSP/DSP Linker Specifications (ST-70)

5. Sound Development Manual Outline (ST-81)

SEGA OF AMERICA, INC.
Consumer Products Division

SCSP Waveform Editor Technical Specifications

Doc. # ST-67-121593

This software is a waveform editing tool that has three main functions:

- Waveform Retrieval
- Waveform Editing
- Waveform Output

1.0 Waveform Retrieval

Waveforms may be retrieved from a file or the SCSP. The retrieved waveform is displayed on-screen.

Retrieving Waveforms from a File

This program can edit waveforms retrieved from Audio Interface File Format (AIFF) files stored on a Macintosh computer. The AIFF format consists of a header and wave data. The header contains data such as number of channels, number of samples, sampling rate, sample bit resolution, loop start point, loop end point, and base note.

Sampling Waveforms with the SCSP

This editor can also sample sounds onto a hard disk through the SCSP's *Digital In* terminal. The data that is retrieved from the SCSP is written onto the Macintosh hard disk and can be displayed on-screen for editing.

2.0 Waveform Editing

The waveform that is displayed on-screen can be edited in many ways. The editing functions include:

- Cut & paste
- Waveform mixing
- Looping (editing of loop start and loop end points)
- Single-byte level wave data editing
- Resampling
- Pitch shifting
- Waveform Scaling
- Low-pass, high-pass, and band-pass filting
- Crossfade, fade in, fade out

3.0 Waveform Output

This editor saves the modified wave data in the Macintosh as AIFF data. It can also play back the saved data using the SCSP or the Macintosh. The following data is saved in the file:

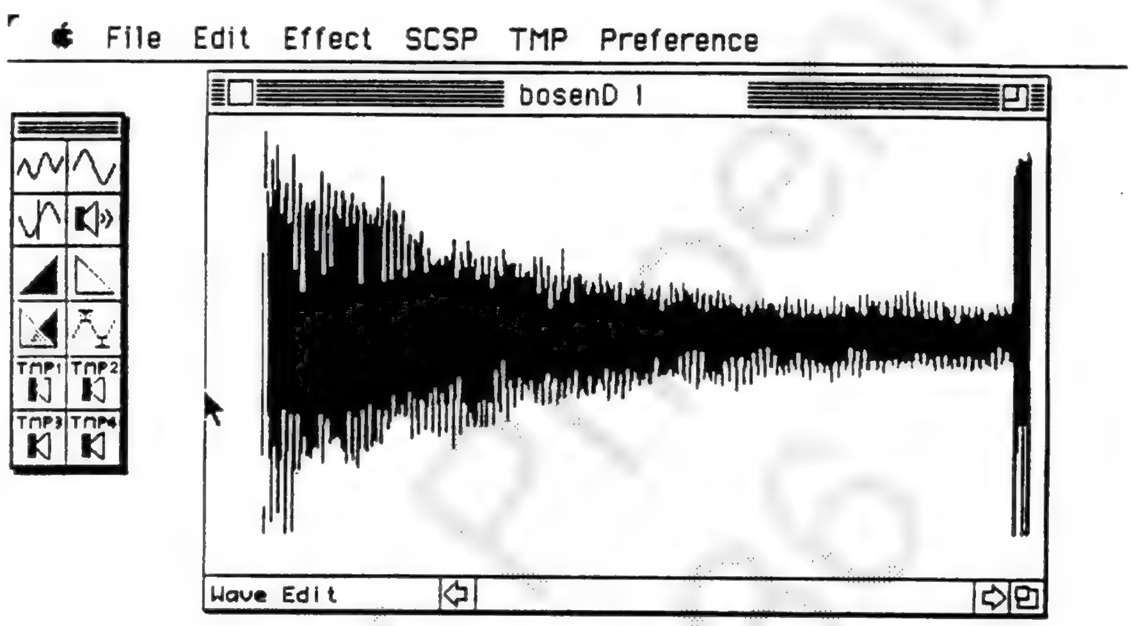
- Wave data
- Sampling rate
- Number of samples
- Sample bit resolution
- Loop start point
- Loop end point
- Note
- Loop mode

4.0 Starting the Editor

From the Finder, double-click the icon for the Waveform Editor. The system downloads the 68000 program used by the Editor into the target and then executes it. If the program has already been downloaded, (For example, by the prior use of the Waveform or Sound Editors) the download is not performed. The ProgramReady flag determines whether or not the program is downloaded..

After downloading and executing the program, the system starts the Waveform Editor. When the tool is started, the menu and the control window are displayed.

What Is Seen on Screen



The Editor's screen layout is shown in the figure above. The basic operations of this software are managed through menus, a Control Window (window on the left), and an Edit Window (window on the right). The use of these components are described in the following sections. (Please note that this document does not describe basic Macintosh operations, such as working with windows.)

5.0 Control Window

The Control Window is available for the quick execution of frequently used menu items by pointing and clicking the mouse. Clicking the icons is the same as performing the following menu operations:

Zoom out

Reduces (zooms out) the size of the waveform display when the Wave Edit screen is in the Edit Window. When the active Edit Window is not the Wave Edit screen, using this function switches the window to the Wave Edit screen. This software supports up to 32,768 ZoomIn and ZoomOut levels.

Zoom in

Enlarges (zooms in) the size of the waveform display when the Wave Edit Screen is in the Edit Window. When the Edit Window is not the Wave Edit screen, using this function switches the window to the Wave Edit screen. This software supports up to 32,768 ZoomIn and ZoomOut levels.

Loop Edit

Switches the Edit Window to the Loop Edit screen.

Play Audio

Uses the Macintosh or the SCSP to play back the waveform being edited. If a section of a waveform is selected, this function plays only the selected portion. Looping does not occur during the selective waveform playback.

Fade in

Fades in the selected portion of the waveform. The beginning of the selected portion is set as 0% and the end as 100%. Linear scaling is then performed between the two points.

Fade out

Fades out the selected portion of the waveform. The beginning of the selected portion is set as 100% and the end as 0%. Linear scaling is then performed between the two points.

Cross Fade

Crossfades a selected portion of the waveform and a waveform in the Clipboard. The selected portion of the waveform is faded in and the waveform in the Clipboard is faded out.

Scale

Scales the selected portion of the waveform. When selected, a dialog box is displayed. Enter a scaling rate (0 to 200%) to execute scaling. When the scaling method is 100%, the largest data of the selected waveform is replaced with the maximum value for the bits (127 for 8-bit data and 32,767 for 16-bit data). The rate is also assigned to other data. The other percentages are based on the 100% data. If 100% is exceeded, the maximum value for the bits is exceeded. In this case, however, the value is replaced with the maximum value for the bits.

TMP1Play

Plays back the data stored in the TMP1 file. The playback method is the same as that of the PlayAudio function.

TMP2Play

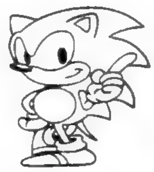
Plays back the data stored in the TMP2 file. The playback method is the same as that of the PlayAudio function.

TMP3Play

Plays back the data stored in the TMP3 file. The playback method is the same as that of the PlayAudio function.

TMP4Play

Plays back the data stored in the TMP4 file. The playback method is the same as that of the PlayAudio function.

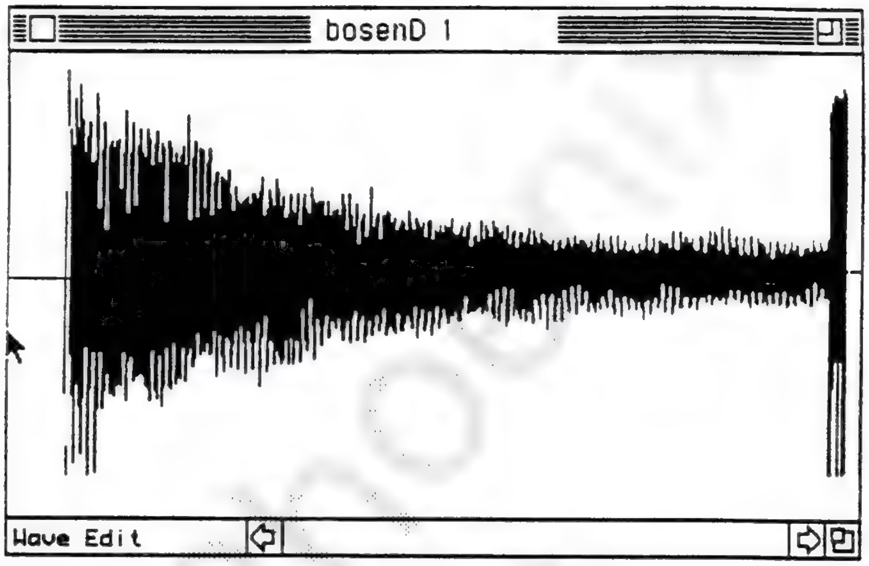


6.0 Edit Windows

This software has three Edit Windows:

- Wave Edit Window
- Loop Edit Window
- Hex Edit Window

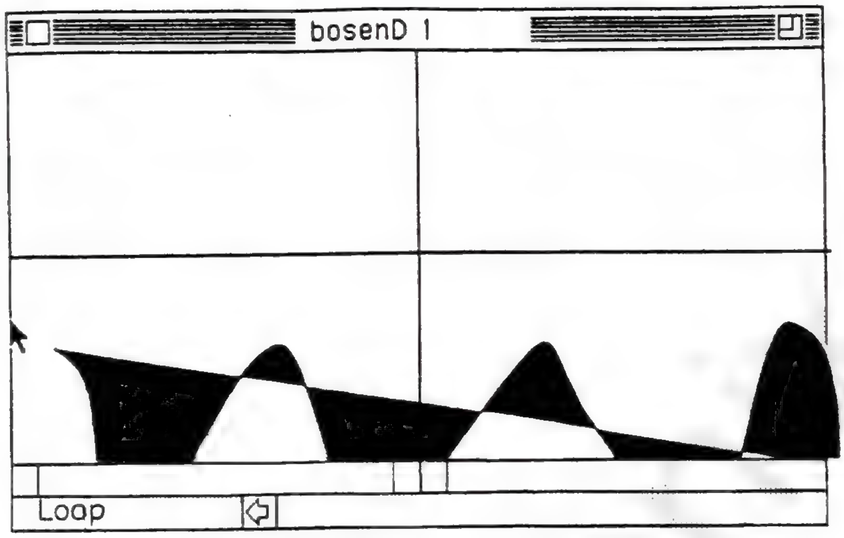
Wave Edit Window



The layout of the Wave Edit Window is as shown in the figure above. The title bar displays the name of the wave. At the bottom of the screen are the title of the Wave Edit screen and the control bar for waveform scrolling. This screen also shows the number of data samples for this waveform, the loop start cursor, and the loop end cursor.

The main function of the Wave Edit Window is to select a waveform (other edit functions are accessed via the menus), change the loop start and loop end points, and scroll through the waveform. All of these operations are executed at the zero cross point.

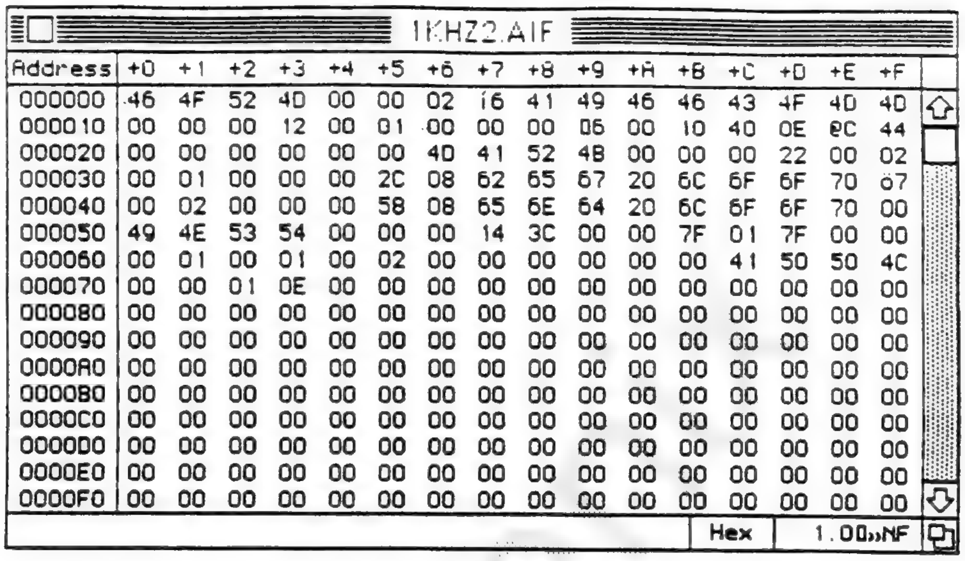
7.0 Loop Edit Window



The layout of the Loop Edit Window is as shown in the figure above. The title bar displays the name of the wave, the title of the Loop Edit Window, and the control bar for waveform scrolling. The screen also shows the amount of the data used by the waveform. The end of the waveform in the left side of the window is the loop end point. The beginning of the waveform in the right window is the loop start point. Make fine adjustments to the loop point by scrolling through the right/left windows to move the loop point. The end loop point (rightmost point in the left window) can be scrolled from the start loop point to the end of the waveform. The start loop point (leftmost point in the right window) can be scrolled from the beginning of the waveform to the end loop point.

The main function of the Loop Edit Window is to give control over the loop start and end points by the use of the scroll bar (other edit functions are accessed via the menus). These operations are performed at each respective point. This window does not allow waveform selection or scrolling, since waveform scrolling would also shift the loop points.

8.0 Hex Edit Window



1KHZ2.AIF

Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F
000000	46	4F	52	4D	00	00	02	16	41	49	46	46	43	4F	4D	4D
000010	00	00	00	12	00	01	00	00	00	06	00	10	40	0E	6C	44
000020	00	00	00	00	00	00	4D	41	52	4B	00	00	00	22	00	02
000030	00	01	00	00	00	2C	08	62	65	67	20	6C	6F	6F	70	67
000040	00	02	00	00	00	58	08	65	6E	64	20	6C	6F	6F	70	00
000050	49	4E	53	54	00	00	00	14	3C	00	00	7F	01	7F	00	00
000060	00	01	00	01	00	02	00	00	00	00	00	00	41	50	50	4C
000070	00	00	01	0E	00	00	00	00	00	00	00	00	00	00	00	00
000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Hex 1.00KNF

The layout of the Hex Edit Window is as shown in the figure above. The title bar displays the name of the wave. The title of the Hex Edit WIndow appears at the bottom of the screen, and the control bar for data scrolling at the right. The screen also shows the amount of data used by the waveform. This window allows the editing of waveform data as hexadecimal data. The left side of the window lists the waveform addresses in sequence from the beginning. The values at the top of the window are the offsets from the addresses, and the main section of the window displays the hex data. The loop start point and the loop end point are displayed in bolface type (the loop end point follows the loop start point).

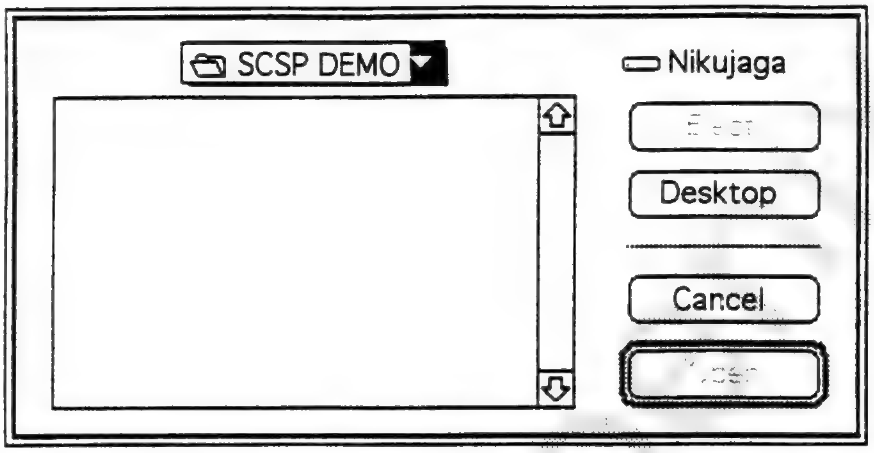
9.0 Menus

File Menu

New

Opens an empty Wave Edit window. The menu item is for sampling a waveform from the SCSP.

Open...

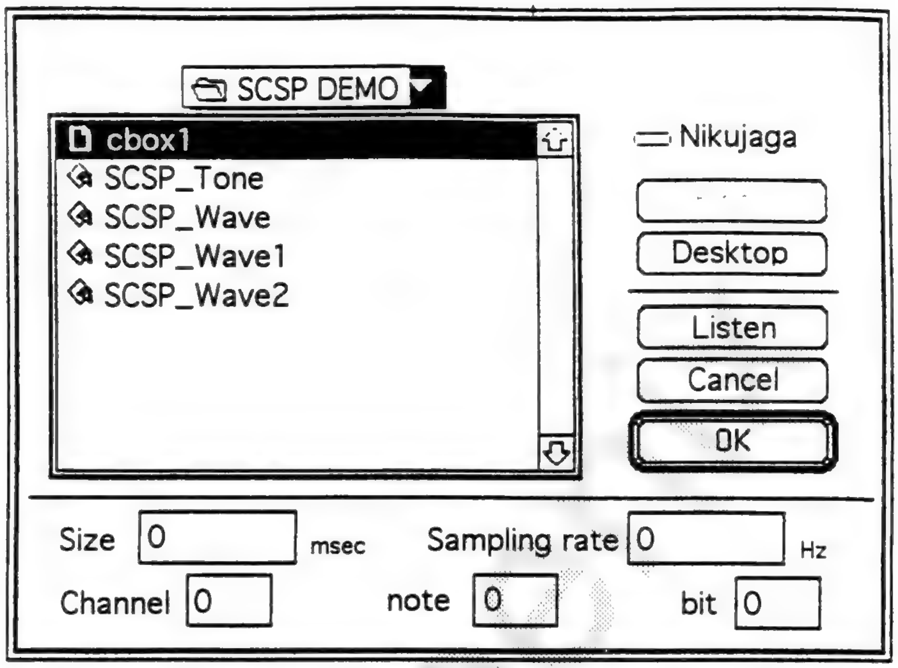


This function reads a waveform from any Macintosh Audio Interface File Format (AIFF) file, regardless of the original application (Alchemy ,Sound Designer II, and so on) that was used to create the file. Valid data includes the wave data, as well as the number of channels, the number of samples, the sampling rate, the sample bit resolution, the loop start point, the loop end point, and the base note.

The above data is used by this Editor, and is displayed in the Wave Edit Window. Other data items are described below.

- Number of channels: The software recognizes one or two channels. For two channels, the software displays two Wave Edit Windows and sets the window titles as *file-name*-L and *file-name*-R.
- Number of samples: The Editor uses this data item when it refers to the number of bytes used by the waveform. For 8-bit data, the number of bytes equals the number of samples. For 16-bit data, the number of bytes is twice the number of samples.
- Sampling rate: The application uses this value when it resamples. When the Resampling dialog box is displayed, this rate is the initial value that is automatically entered into the Now field.
- Sample bit resolution: The application uses this data item to determine whether the wave data is 8-bit or 16-bit data.
- Loop start point and loop end point: The application refers to these values in the Wave Edit and Loop Edit Windows. In the Wave Edit Window, the cursor is placed at the beginning of the wave data specified by these points. In the Loop Edit Window, the waveform is displayed so that the loop end point is located on the right side of the left window, and the loop start point is on the left side of the right window.

Open Special...



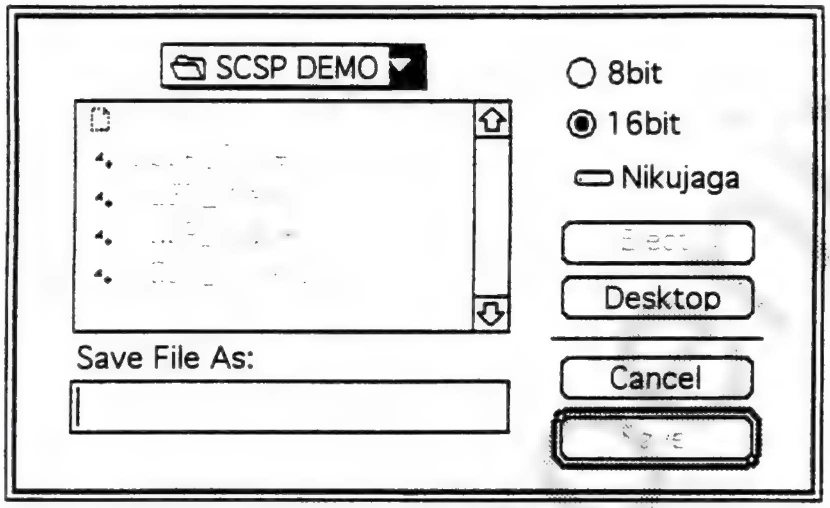
This menu item is an expanded version of the **Open...** menu. When selecting a file, use this menu to view the AIFF header data. The data items displayed here are:

- Size: Number of samples in the data
- Sampling rate: Sampling frequency
- Channel: Number of channels
- note: The waveform's frequency shown as a MIDI note number.
- bit: Sample bit resolution of the wave data

Save

Saves the wave data displayed in the currently active window to a file. The data is saved to the file name that was specified when the file was opened. The following data is saved: wave data, number of channels, number of samples, sampling rate, sample bit resolution, loop start point, loop end point, and base note. If a new Edit Window is opened using the **New...** command, **Save** operates in the same manner as **Save as....**

Save as...



Saves the wave data displayed in the currently active window to a file. The data is saved to a new file specified in the **Save as...** dialog box. The dialog box provides the option of saving the data as 8-bit or 16-bit data. Select the button specifying the desired number of bits and save the data.

Close

Closes the currently active window. If the wave being edited in the window was edited since the wave was opened or saved, a dialog box gives you the option of saving the data before the window is closed.

- Cancel: Cancels the close operation.
- Don't Save: Closes the window without saving the data.
- Save: Executes Save (see the description of the Save command above), then closes the window.

Quit

Quits the Editor. If Edit Windows are open when this option is selected, the system executes **Close** for all Edit Windows (see the description of the **Close** command above). It then terminates the application.

Edit Menu

Undo

Cancels the previous operation. The operations that can be canceled are: modification of the loop start or loop end point, resampling, pitch shift, scaling, filtering, fade in, fade out, crossfade, mix, cut, copy, paste.

Cut...

Cuts the selected waveform data and saves it to the Clipboard. This function can be used when the active window is either the Wave Edit or the Hex Edit Window. When you select this function, a dialog box provides the option to optimize (pack) the vacant area left by the cut, or fill it with zeros.

Cut other

Cuts the waveform data that was not selected. The cut data is not saved to the Clipboard. This function can be used when the active window is the Wave Edit or Hex Edit Window.

Copy

Copies the selected waveform data and saves it to the Clipboard. This function can be used when the active window is the Wave Edit or Hex Edit Window.

Paste...

Inserts the data saved in the Clipboard. The Clipboard data is inserted at the cursor position. This function can be used when the active window is the Wave Edit or Hex Edit Window. When this function is selected, a dialog box opens asking whether to insert the Clipboard data at the paste position or overwrite the existing data instead.

Zoom in

Enlarges the size of the waveform display with the cursor as the center of magnification. This function can be used when the active window is the Wave Edit Window.

Zoom out

Reduces the size of the waveform display with the cursor as the center of magnification. This function can be used when the active window is the Wave Edit Window.

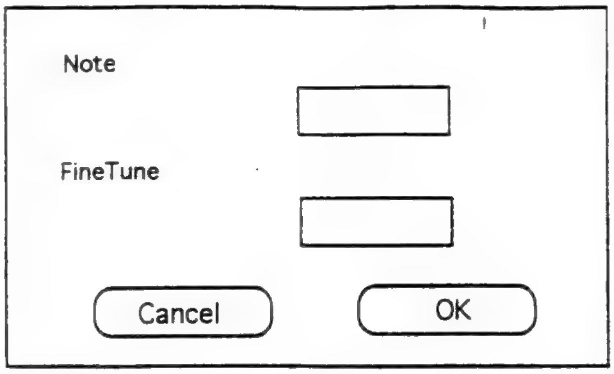
Select all

Selects all waveform data. This function can be used when the active window is the Wave Edit or Hex Edit Window.

Select loop

Selects the loop portion of the waveform data. The loop portion is the data between the loop start and loop end points. This function can be used when the active window is the Wave Edit or Hex Edit Window.

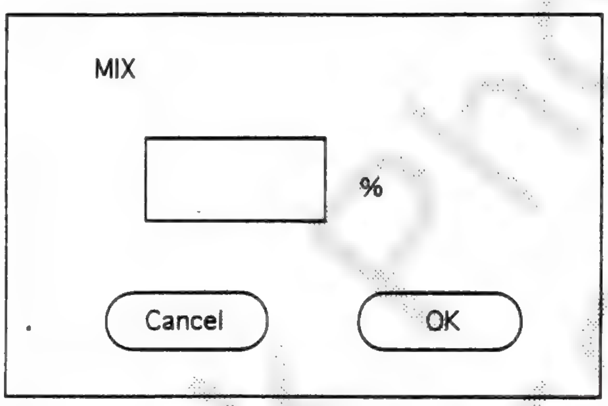
Note...



This function opens the dialog box shown in the figure above. This dialog box shows the current note and fine tune values. To change a value, enter a new value and click the OK button.

Mix...

Mixes the waveform stored in the Clipboard with the waveform data being edited and displays the resulting waveform. The waveform stored in the Clipboard is mixed in with the cursor location as the starting point of the mix.



Enter the mix rate as a percentage. The possible values are 0 to 100%

<u>Mixing Method</u>

The mixing method employed by this Editor treats the waveforms as data when mixing waveforms. Thus, it does not recognize differences in sampling rates. The data displayed in the window determines the bit resolution. If the waveform in the window has 8-bit data and the mix file has 16-bit data, the lower 8 bits of the mix file data are deleted. If the window has 16-bit data and the mix file has 8-bit data, the mix file data is increased 256-fold before being mixed. If the data entered in the MIX field of the dialog box is M, the file wave data as Xn, and the window wave data as Yn, the data after mixing, Zn, is as follows:

```
Zn = Xn * M/100 + Yn * (100 - M)/100
```

Differences in the number of samples between the Clipboard wave data and the window wave data are processed as follows:

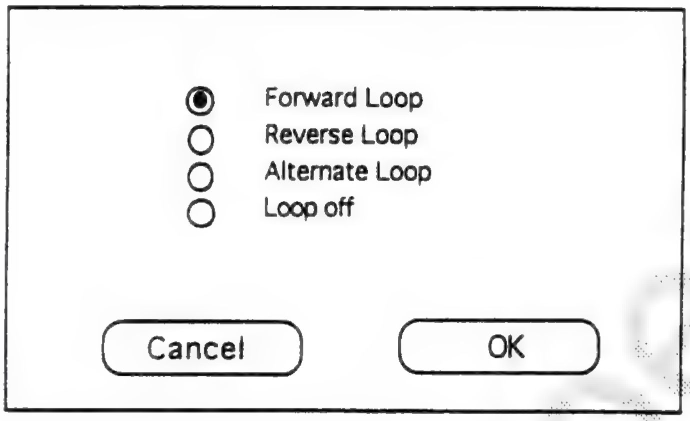
- If the Clipboard wave data is longer: Mixing stops where the wave data in the window ends.
- If the window wave data is longer: Mixing continues until the end of the file data. The subsequent data remains unchanged.

Wave Edit
Changes the active window to the Wave Edit Window.

Loop Edit
Changes the active window to the Loop Edit Window.

Hex Edit
Changes the active window to the Hex Edit Window.

Loop Mode...

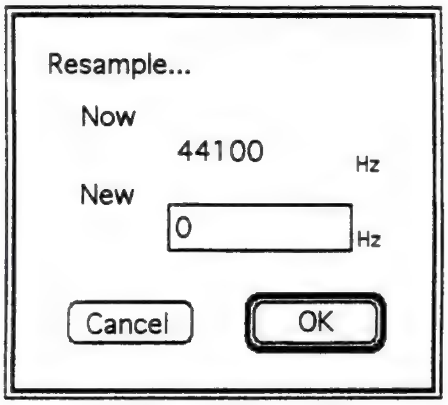


Allows the selection of the loop mode to be used when wave data is played with the Play Audio function. The Alternate Loop option cannot be used for playback on the Macintosh.

Effect Menu

The Effect menu can be operated when the active menu is the Wave Edit or the Hex Edit Window.

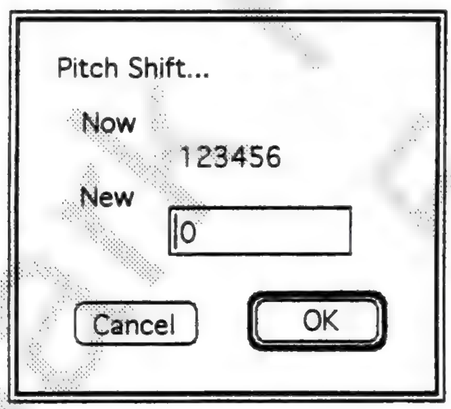
Resample...
Resamples the waveform being edited.



The value displayed in the Now field is the current sampling rate. This rate is the sampling rate included in the AIFF header or the sampling rate that was converted by a previous resampling operation.

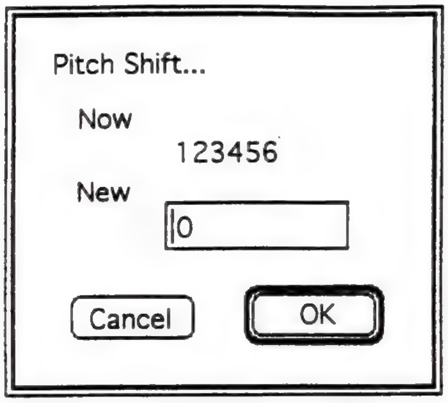
The values that can be entered in the New field are 1 to 65,535 hertz. (The resampling method is explained in a separate document.)

Pitch Shift...
Changes the pitch of the waveform being edited.



The value displayed in the Now field is the current pitch (note). This pitch is the note included in the AIFF header or the note that was converted by a previous pitch shift operation. The values that can be entered in the New field are 1 to 127. (The pitch shift method is explained in a separate document.)

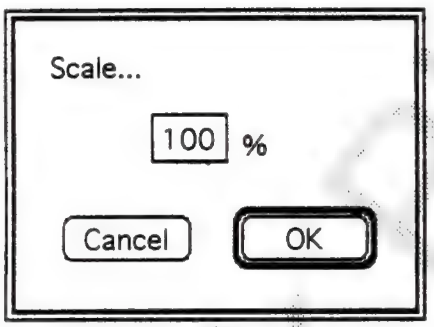
Size Shift...



The value displayed in the **Now** field is the current size. This value is the size included in the AIFF header or the size that was produced by a previous size shift. (The size shift method is explained in a separate document.)

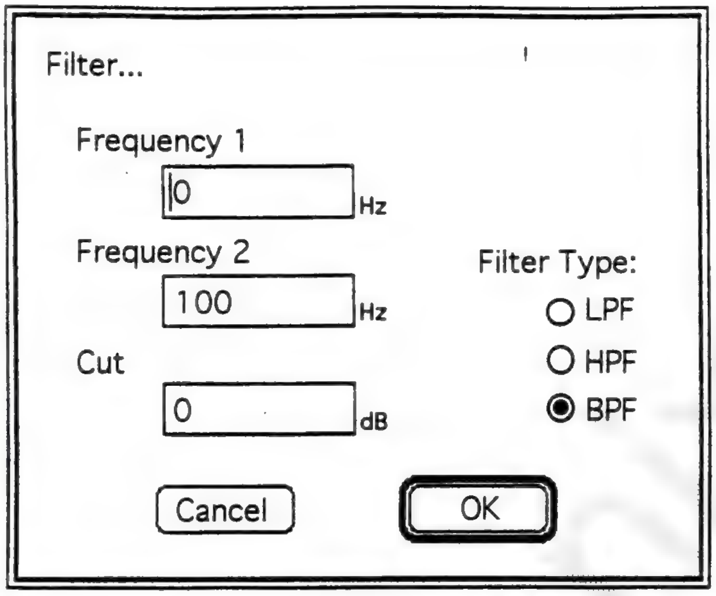
Scale

Scales the selected portion.



Enter a scaling rate (0 to 200%) to execute scaling. When the scaling method is 100%, the largest data of the selected waveform is replaced with the maximum value for the bits (127 for 8-bit data and 32,767 for 16-bit data). The rate is also assigned to other data. The other percentages are based on the 100% data. If 100% is exceeded, the maximum value for the bits is exceeded. In this case, however, the value is replaced with the maximum value for the bits.

Filter...



Executes low-pass, high-pass, or band-pass filtering. Use the radio buttons to select a filter type and enter data in the following fields:

- Frequency 1 and Frequency 2 (1 to 65,535): Frequency at which filter cutoff processing starts. Frequency 2 is used for band-pass filtering but not for the other filter types.
- Cut (1 to 255): Enter in decibels the volume at which sounds are lowered per octave.

(The filtering methods are explained in a separate document.)

CrossFade
Executes a crossfade between a selected portion of the waveform and a waveform in the Clipboard. The crossfade method fades in the selected portion of the waveform and fades out the waveform in the Clipboard. For information on the fade in and fade out methods, see the explanations below.

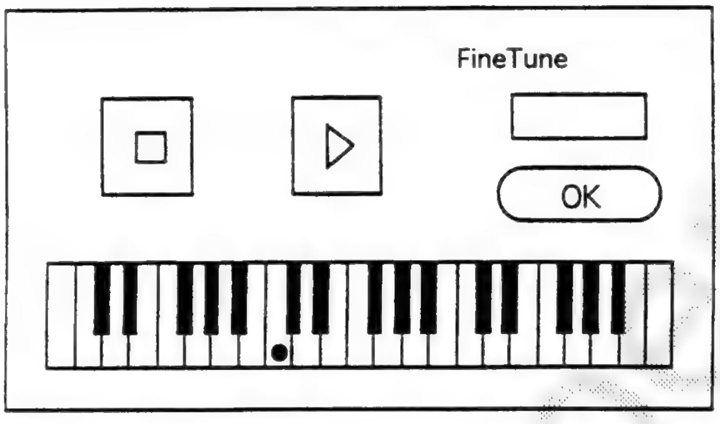
Fade in
Fades in the selected portion of the waveform. The fade in method sets the beginning of the selected portion as 0% and the end as 100%. Linear scaling is then performed between the two points.

Fade out
Fades out the selected portion of the waveform. The fade out method sets the beginning of the selected portion as 100% and the end as 0%. Linear scaling is then performed between the two points.

SCSP Menu

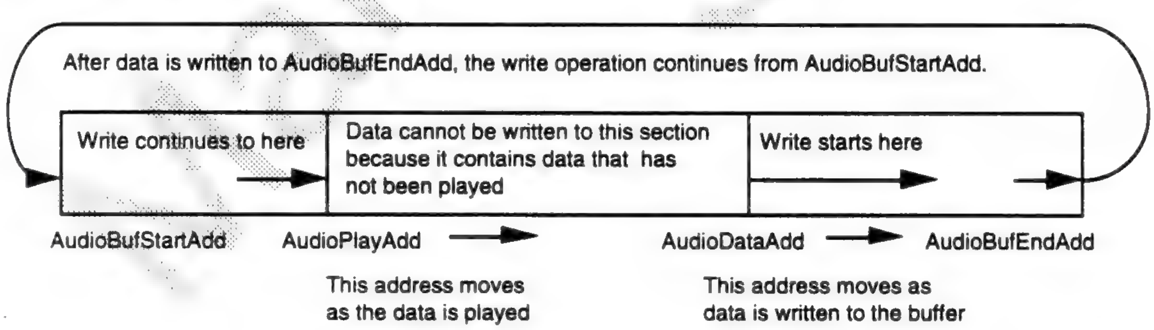
Play Audio

Uses the Macintosh or the SCSP to play the waveform being edited. If a waveform is selected, this option plays only the selected portion. Loop playback is not executed during selective waveform playback. If the selected waveform or the entire waveform is in the 68000's audio buffer (size of which to be determined later), a keyboard is displayed on-screen for playback with pitch control. Clicking the desired key on the keyboard will place a dot on the key and change the pitch of the sample for playback. Fine tuning can be changed by using this function as well.



SCSP Audio Play Back Explanation

The Waveform Editor first changes the 68000's task processing to execute **Play Audio** by writing "PlayAudio" into the 68000's Command memory address. The application then checks AudioDataAdd in the memory addresses to find out how much data has accumulated in the 68000's audio playback buffer. It then writes data from the AudioDataAdd position. When the address of the write data reaches AudioBufEndAdd (the buffer end address), the application goes to AudioBufStartAdd (the buffer start address) and starts writing data from that position. When the address of the write data reaches AudioPlayAdd (the address of the data currently being played), the application stops write processing since the buffer is full. When buffer space becomes available, write processing starts again. This process uses the buffer as a ring buffer.



Audio buffer

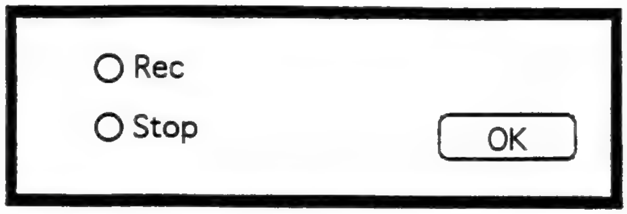
As explained earlier, the SCSP plays back wave data as the Editor writes data in the 68000's audio buffer. Loop processing is handled by the Editor because the SCSP's loop processing can only process the waveform in the memory area allocated to the SCSP.

Memory Playback Explanation

When PlayAudio is selected, the Waveform Editor performs the following processes while it retrieves the waveform into the audio buffer.

- Data is written from the start of the audio buffer.
- The SCSP hardware plays back the waveform.
- Slot 0 is used.
- KYONB is set to "1" for slot 0, and then sets SA, LSA, and LEA. SA is AudioBufStartAdd, LSA is the AIFF BeginLoop, and LEA is the AIFF EndLoop.
- For 8-bit data, the application writes "1" to PCM8B.
- LPCTL inputs loop information.
- "31(1FH)" is written to AR and RR.
- When a key on the on-screen keyboard is clicked, the application determines OCT and FNS by the number of half steps between the key and the sampled note (see table). The application then sets KYONEX to "1" and plays the sound. When the mouse button is released, the application sets KYONEX to "0" and turns off the sound.

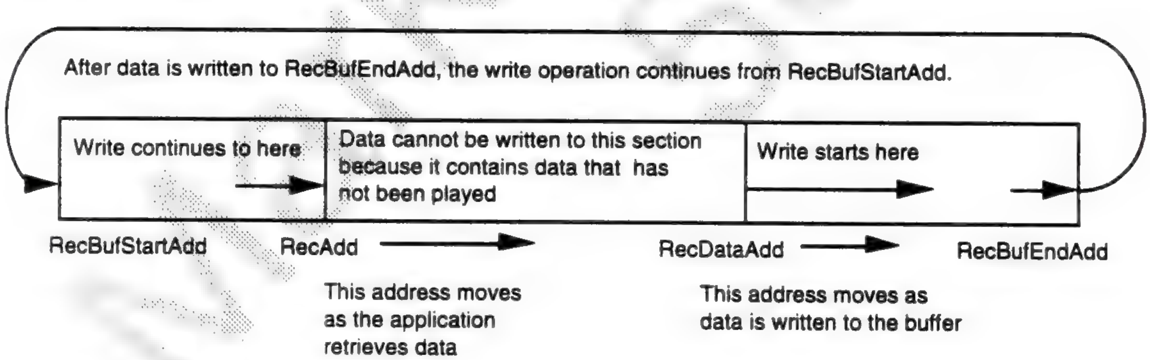
Get Sound
Discards the currently active wave data, saves the data from the SCSP onto the hard disk, and displays the waveform in the window. When this function is selected, a dialog box is displayed.



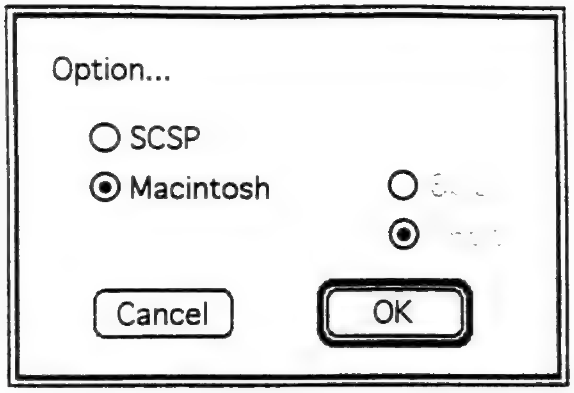
Tentative version

If the Cancel button is pressed at this point, the Waveform Editor terminates the hard disk recording process without discarding the waveform that was being edited. If the **Rec** button is pressed, the tool starts hard disk recording. If the **Stop** button is pressed, the editor stops hard disk recording. The waveform data remains on the hard disk as TEMP. If the **OK** button is pressed, the Editor displays this waveform in the previously active window for editing.

Hard Disk Recording Explanation
When the **Rec** radio button is pressed, the Waveform Editor changes the 68000 processing to hard disk recording by writing "RecAudio" to the 68000's Command memory addresses. As soon as the application confirms the RecAudio command, it writes data to the 68000's audio recording buffer at each 44.1KHz interval. The application then checks the RecDataAdd address to find out the amount of data that has been accumulated. The application retrieves data starting from RecAdd, the first address of the previously read data, and writes data to the hard disk. (Initially, the application retrieves data from RecBufStartAdd, the first address of the buffer.) During this operation, the application continues to store data into the buffer. When RecDataAdd reaches RecBufEndAdd, the last address of the buffer, the application goes to RecBufStartAdd and starts writing data from that position. This process uses the buffer as a ring buffer.



Option...



Selects the output device used by the **Play Audio** function to play back waveform data. The output choices are Macintosh and SCSP. Either 8-bit or 16-bit playback is possible when using SCSP. When the Macintosh option is selected, playback using the **Alternate Loop** mode is disabled.

TMP Menu

Stock TMP1

Use the **Stock TMP1** command to compare the original and edited sounds. This command saves the wave data being edited to a temporary AIFF-format file called *TMP1*. To hear the original sound while editing this data in the Edit Window, execute the **Play TMP1** command. The TMP1 file is deleted when the application is terminated.

Stock TMP2

Use the **Stock TMP2** command to compare the original and edited sounds. This command saves the wave data being edited to a temporary AIFF-format file called *TMP2*. To hear the original sound while editing this data in the Edit Window, execute the **Play TMP2** command. The TMP2 file is deleted when the application is terminated.

Stock TMP3

Use the **Stock TMP3** command to compare the original and edited sounds. This command saves the wave data being edited to a temporary AIFF-format file called *TMP3*. To hear the original sound while editing this data in the Edit Window, execute the **Play TMP3** command. The TMP3 file is deleted when the application is terminated.

Stock TMP4

Use the **Stock TMP4** command to compare the original and edited sounds. This command saves the wave data being edited to a temporary AIFF-format file called *TMP4*. To hear the original sound while editing this data in the Edit Window, execute the **Play TMP4** command. The TMP4 file is deleted when the application is terminated.

Play TMP1

Plays the data stored in the TMP1 file. The playback method is the same as in the Play Audio function.

Play TMP2

Plays the data stored in the TMP2 file. The playback method is the same as in the Play Audio function.

Play TMP3

Plays the data stored in the TMP3 file. The playback method is the same as in the Play Audio function.

Play TMP4

Plays the data stored in the TMP4 file. The playback method is the same as in the Play Audio function.

Revert TMP1
Restores the data stored in the TMP1 file to the currently active window. The previous data in the window is erased.

Revert TMP2
Restores the data stored in the TMP2 file to the currently active window. The previous data in the window is erased.

Revert TMP3
Restores the data stored in the TMP3 file to the currently active window. The previous data in the window is erased.

Revert TMP4
Restores the data stored in the TMP4 file to the currently active window. The previous data in the window is erased.

SEGA

SEGA OF AMERICA, INC.
Consumer Products Division

Technical Specifications for the Sound Editing Tool

Doc. # ST-68-121593

1.0 Introduction

This software is a tool for editing sound parameters used by the SCSP. The main functions of this tool control:

- Real-time sound playback/editing
- Mixer settings
- Sound monitoring
- Sound format editing

Real-time Sound Playback/Editing

While this application is running, the target receives MIDI data from an external source and produces sounds. When producing the sounds, the target refers to the sound data created by this application. This enables sound editing while sounds are being played and monitoring of the edits in real time.

Mixer Settings

This application will enable access of the SCSP's mixer function and edit mixer data and save it to a data file.

Sound Monitoring

Because this application can play sounds while sounding the target, this application can be used to monitor sounds. It can also view data thought to be necessary for the sound being reproduced.

Sound Format Editing

The SCSP can play music via MIDI. Sound data is referenced for MIDI playback. This data is organized based on a sound format, and the target's program produces sounds by referring to this format. The sound edit tool can edit sound-format data and save the edited data to a file. The sound format is made up of the following components:

- Sound parameters that correspond to MIDI program numbers.
- Mixer data (including master volume).
- Percussion parameters.

The sound edit tool can also edit these types of data and save the edited data to a file.

2.0 File Formats

The sound editing tool supports two file formats: the Macintosh-based SCSP format that contains detailed information, and the SCSPBIN format, which contains only the data that is placed into the 68000's memory along with header data necessary to create a Macintosh file.

SCSP Format

The SCSP format is as follows:

ckID	'SCSP'	4 Bytes
ckSize	176516	4 Bytes
ckType	'scsp'	4 Bytes
formType	'voce'	4 Bytes
	MixerChunk	
	VoiceChunk	

The format begins with 'SCSP' ID, which is four bytes of ASCII code. This is followed by the total number of bytes for the Mixer Chunk, Voice Chunk, and Percussion Chunk. This variable is long and takes up four bytes. Next is 'voce', which is four bytes of ASCII code. After that are the Mixer Chunk, the Voice Chunk, the Percussion Chunk, and the data.

Mixer Chunk

ckID	'MIXR'	4 Bytes
ckSize		4 Bytes
	Mixer0	
	Mixer1	
	• • •	
	Mixer L	

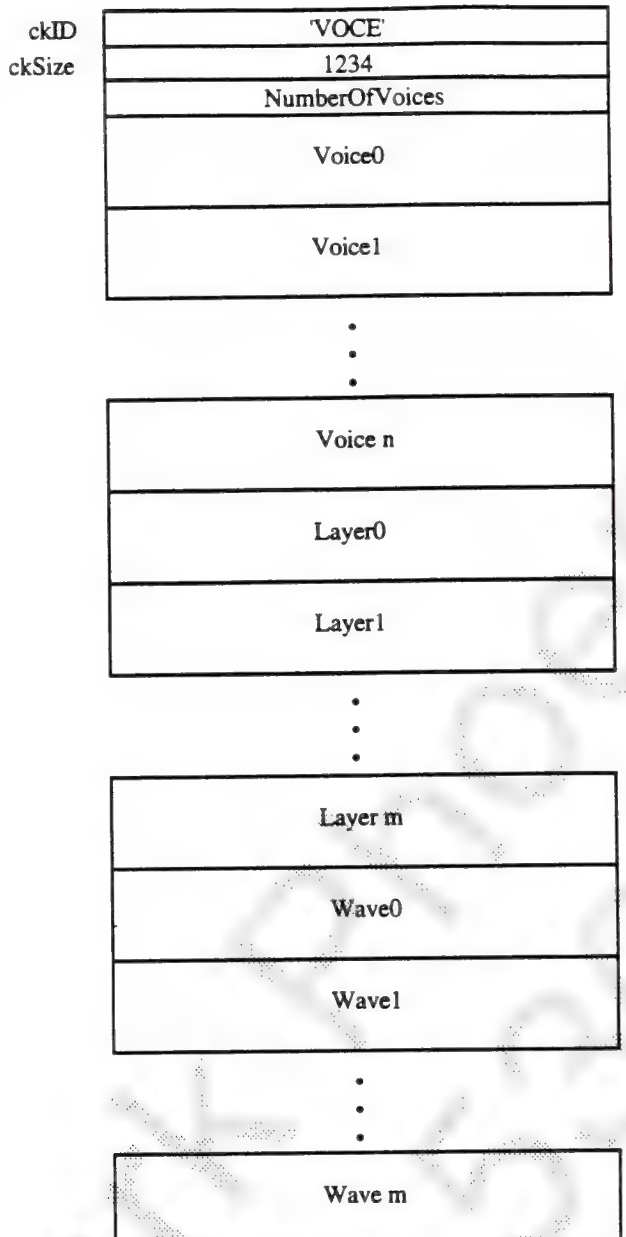
The Mixer Chunk consists of the data for 16 mixer channels and a header. The header ID is 'MIXR', which is four bytes of ASCII code. Next, the number of bytes in the mixer data is entered with four bytes. In this revision, the Mixer Chunk has been modified to include multiple mixers.

Mixer

EFSDL0[2:0]	EFPAN0[4:0]	1 byte
EFSDL1[2:0]	EFPAN1[4:0]	1 byte
:		
EFSDL17[2:0]	EFPAN17[4:0]	1 byte

Bits 0 to 4 are the PAN data; bits 5 to 7 are SendReturn. This configuration is the same as the SCSP register configuration. Slot 0 corresponds to 100017H.

Voice Chunk



The Voice Chunk consists of Layer data containing Voice parameters (including PCM data) that form the Voice and a header. The header ID is 'VOCE', which is entered in ASCII code and is four bytes. Next, the total number of bytes in the Voice, Layer, and Wave (PCM) data is entered in four bytes. The number of Voices (number of sounds) is as follows.

Voice

VoiceName	16 bytes
BendRangeWidth (H, 6 bits) BendRangeWidth (L, 8 bits)	2 bytes
PlayMode	1 byte
NumberOfLayers	2 bytes
VolBias	1 byte
StartNote EndNote LayerNumber	
StartNote EndNote LayerNumber	
• • •	

The Voice data is displayed above. First, the number of Layers included in the Voice is set at the head of the data. This value is determined by the number of FM operators (key splits). For two operators, the number of Layers is two. For four operators, the number of Layers is four. Next are the Voice name, which is 16 bytes of ASCII code, and a Layer number (including start note, and end note) for each of the specified number of Layers. The Layer numbers can be from 0 to 65,535. These Layer numbers are referenced by the Layer data that follows the Voice data.

- **VoiceName:** Specifies the name of the Voice used by this software. The name is entered with 16 bytes and displayed in the Voice window.
- **NumberOfLayers:** Specifies the number of Layers used by the Voice.
- **VolBias:** Specifies the volume adjustment. The allowed values are complements of two.
- **BendRangeWidth:** This is the data in the Voice window. The data is 14 bits. These two bytes contain the highest six bits and the lowest eight bits.
- **PlayMode:** Specifies legato, portamento, or mono mode. When bit 0 is 1, the play mode is mono mode. When bit 1 is 1, the play mode is legato mode. When bit 2 is 1, the play mode is portamento mode.
- **LayerNumber:** Specifies the Layer number of a Layer used by this Voice.
- **StartNote:** Specifies the note number of the first note produced by this Layer.
- **EndNote:** Specifies the note number of the last note produced by this Layer.

Layer

LayerName	16 bytes
WaveNumber	
WaveSize	4 bytes
SCSP Register0 : SCSP Register22	
BaseNote	
FineTune	
MDXSL	
MDYSL	
VelocityPoint0 VelocityLevel0	
VelocityPoint1 VelocityLevel1	
VelocityPoint2 VelocityLevel2	
VelocityLevel3	
DELAY TIME OFFSET LEVEL ATTACK LEVEL ATTACK TIME DECAY LEVEL DECAY TIME SUSTAIN LEVEL SUSTAIN TIME RELEASE LEVEL RELEASE TIME	
PLFO DELAY TIME DEPTH LEVEL FRQ TIME FADE TIME	

The Layer data includes data that is written to the SCSP registers, information on the frequencies that correspond to MIDI note numbers, and pitch EG data.

- **WaveNumber:** Specifies the number of the Wave used by the Layer.
- **SCSP Registers 0 to 16:** When representing Slot 0, register 0 is 100000H and register 21 is 100015H. However, the FM registers MDXSL and MDYSL are 0. For details, see the parameter window for the FM registers. For 100010 and 100011H, bit 15 is set to 1 when the switch of the Slot window is ON. For 100000 and 100001H, bit 14 is set to 1 when PLFOON is checked, and bit 15 is set to 1 when the PEG is checked. The start address contains the data used by the SCSPBIN format.
- **BaseNote:** Specifies the MIDI note number for the Wave data frequency used by the Layer.
- **FineTune:** Specifies the fine adjustment for the Wave data frequency used by the Layer. The allowed values are 0 to 127, with 64 corresponding to a fine adjustment of 0. For values below 64, the pitch is lowered. For values above 64, the pitch is raised. Each increment changes the pitch by 1/128 of a half step (one MIDI note/128). This parameter is used for final pitch correction and for creating effects such as chorusing.
- **MDXSL:** Bits 0 to 6 of this register specify the Layer number(s) connected via FM for this Layer. Bit 7 specifies the generation.
- **MDYSL:** Bits 0 to 6 of this register specify the Layer number(s) connected via FM for this Layer. Bit 7 specifies the generation.
- **PitchEG Mode:** Specifies the pitch EG loop and the starting time. When bit 7 of the first byte is 1, loop mode is set. When bit 6 is 1, the key-on pitch is the original pitch. When bit 6 is 0, the sound's pitch at key-on is shifted from the original pitch by the offset value, and then returns the pitch back to the original. The second byte of this parameter specifies the delay time. Bit 0 of the first byte is a switch that specifies whether or not this Layer is output as a sound. When this bit is 1, the Layer is used for FM and is not output.
- **VelocityPoint0~2:** Determines the points (velocity numbers) of the velocity curve.
- **VelocityLevel0~3:** Set the levels for each point on the velocity curve.
- **DELAY TIME:** PEG delay time.
- **OFFSET LEVEL:** PEG offset level (pitch offset).
- **ATTACK LEVEL, DECAY LEVEL, SUSTAIN LEVEL, RELEASE LEVEL:** Specify the levels (pitch offsets) for PEG attack, decay, sustain, and release.
- **ATTACK TIME, DECAY TIME, SUSTAIN TIME, RELEASE TIME:** Specify the times to reach PEG attack, decay, sustain, and release.
- **PLFO DELAY TIME:** Specifies the PLFO delay time.
- **FRQ DEPTH LEVEL:** Specifies the change width of the PLFO waveform (pitch offset).
- **FRQ TIME:** Specifies the time for PLFO waveform to reach the change width, starting from 0.
- **FADE TIME:** Specifies the time for fade-in to end.

Wave

WaveName	32 bytes
NumberOfSamples	2 bytes
Bit	
PCM Data	

This data contains items related to the PCM data:

- **WaveName:** The AIFF file name from the waveform's source.
- **NumberOfSamples:** The number of samples in the waveform.
- **Bit:** The bit resolution (8 or 16 bits) of the waveform.
- **PCM Data:** Contains the actual PCM data.

SCSPBIN Format

The SCSPBIN format is as follows:

ckID	'SCSP'	4 bytes
ckSize	54384	4 bytes
ckType	'BIN'	4 bytes
	Mixer offset	2 bytes
	Layer offset	2 bytes
	Voice 0 offset : Voice N offset	2 bytes * N
	Mixer 0 : Mixer L	
	Voice 0 : Voice N	
	Layer 0 : Layer M	
	Wave 0 : Wave K	

The SCSPBIN format does not include the Voice name, Layer name, Wave number, and Wave size that are included in the SCSP format.

- **Mixer offset:** Specifies the offset address from which mixer data begins. The offset address is: 4 [number of bytes in Mixer and Layer offset values] + 2 * (N +1) [number of bytes in Voice offset values]
- **Voice offset:** Specifies the offset address from which Voice data begins. The offset is: 4 + 2 * (N + 1) + 18 * (L + 1) [number of bytes in the mixer data]
- **Layer offset:** Specifies the offset address from which the Layer data begins.

Mixer

EFSDL0[2:0]	EFPAN0[4:0]	1 byte
EFSDL1[2:0]	EFPAN1[4:0]	1 byte
:		
EFSDL17[2:0]	EFPAN17[4:0]	1 byte

Voice

BendRangeWidth (H, 6 bits) BendRange(L, 8 bits)	2 bytes
PlayMode	1 byte
TimeBitOffset	
NumberOfLayers	
VolBias	1 byte
StartNote EndNote LayerOffset	2 bytes
StartNote EndNote LayerOffset	

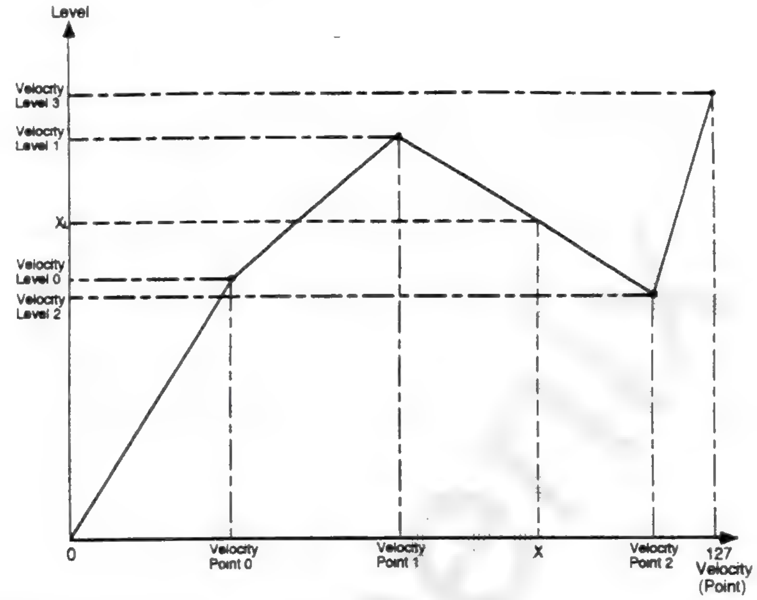
- **TimeBitOffset:** Specifies the bit width offset in the time table used by the PLFO, PEG, and velocity. In the current version, the offset value is normally 3.
- **LayerOffset:** Specifies the offset of the Layer referred to by this section.

Layer

SCSP Register 0
:
SCSP Register 22
BaseNote
FineTune
MDXSL
MDYSL
ApproximationValue0
VelocityPoint0
VelocityLevel0
ApproximationValue1
VelocityPoint1
VelocityLevel1
ApproximationValue2
VelocityPoint2
VelocityLevel2
ApproximationValue3
DLY
OL
AR
AT
DR
DT
SR
ST
RR
RT
Delay
FRQ
FDR
FDT

Approximation values: Specifies the approximation value table numbers used for velocity data processing. These values are obtained from calculations that use velocity points 0 to 3 and velocity levels 0 to 3.

Obtaining the ApproximationValues



The figure above shows the relationship between the velocity points and the velocity levels. First draw a level curve, like the one above, for velocities 0 to 127. Next, calculate the four curve slopes. (The equations for calculating the slopes are shown below.) For each calculated slope, check the approximation value table and find the table number that has the closest value to the calculated slope. Use that number as the approximation number.

ApproximationValue0:

$$\frac{\text{VelocityLevel0}}{\text{VelocityPoint0}}$$

ApproximationValue1:

$$\frac{\text{VelocityLevel1} - \text{VelocityLevel0}}{\text{VelocityPoint1} - \text{VelocityPoint0}}$$

ApproximationValue2:

$$\frac{\text{VelocityLevel2} - \text{VelocityLevel1}}{\text{VelocityPoint2} - \text{VelocityPoint1}}$$

ApproximationValue3:

$$\frac{127 - \text{VelocityLevel2}}{127 - \text{VelocityPoint2}}$$

Approximation Value Table

0	D6	D5	D4	D3	D2	D1	D0

The relationship between the slope and the bits in the approximation value table is as follows:

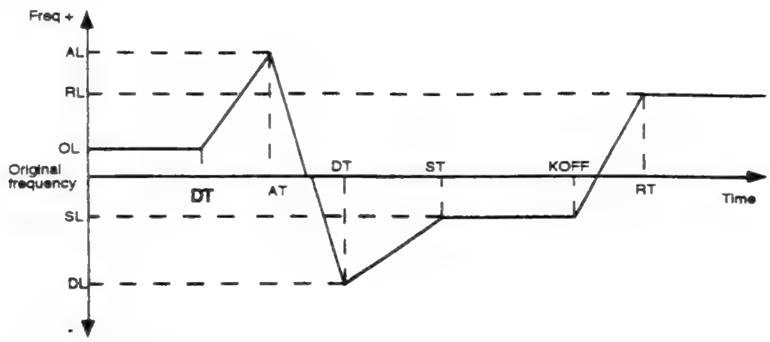
D2 - D0	Slope
0	$\pm\infty$
1	$1 < \text{slope} < +\infty$
2	1
3	$0 < \text{slope} < +1$
4	0
5	$-1 < \text{slope} < 0$
6	-1
7	$-\infty < \text{slope} < -1$

Use the following two tables to look up the approximation values.

D6 - D3	Slope (D2 - D0 = 1 or 7)	Slope (D2 - D0 = 3 or 5)
0	Invalid	Invalid
1	1.5	1/1.5
2	2	1/2
3	3	1/3
4	4	1/4
5	6	1/6
6	8	1/8
7	12	1/12
8	16	1/16
9	24	1/24
A	32	1/32
B	48	1/48
C	64	1/64
D	96	1/96
E	128	1/128
F	Invalid	Invalid

The allowed approximation values are $\pm\infty$, 1, 0, and the table values. From the allowed values, find the value that is closest (smallest absolute difference) to the actual value. Then set D0-2 and D6-3 to the approximation value.

PEG Parameters



OL: OFFSET LEVEL
AL: ATTACK LEVEL
DL: DECAY LEVEL
SL: SUSTAIN LEVEL
RL: RELEASE LEVEL
DT: DELAY TIME
AT: ATTACK TIME
DT: DECAY TIME
ST: SUSTAIN TIME
RT: RELEASE TIME

The slopes (rates) and times as shown in the figure above are incorporated into the SCSPBIN format.

- DLY: Specifies the table number of the time table used for the PEG delay time. The time table lists the number of counts for each unit of time. To obtain the DLY value, first calculate the number of counts from the DELAY TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set this table number as the DLY value.
  Number of counts = DELAY TIME/2 (msec unit time)
- OL: Offset level from the key-on note when key-on was executed.
- AR: Specifies the level change per unit time.
  AR = ATTACK LEVEL/AT
- AT: Specifies the table number of the time table used to control the time it takes to reach the attack level. The time table lists the number of counts for each unit of time. To obtain the AT value, first calculate the number of counts from the ATTACK TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the AT value.
  Number of counts = ATTACK TIME/2 (msec)
- DR: Specifies the level change per unit. DR = DECAY LEVEL/DT
- DT: Specifies the table number of the time table used for the time it takes to reach the decay level. The time table lists the number of counts for each unit time. To obtain the DT value, first calculate the number of counts from the DECAY TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the DT value.
  Number of counts = DECAY TIME/2 (msec)

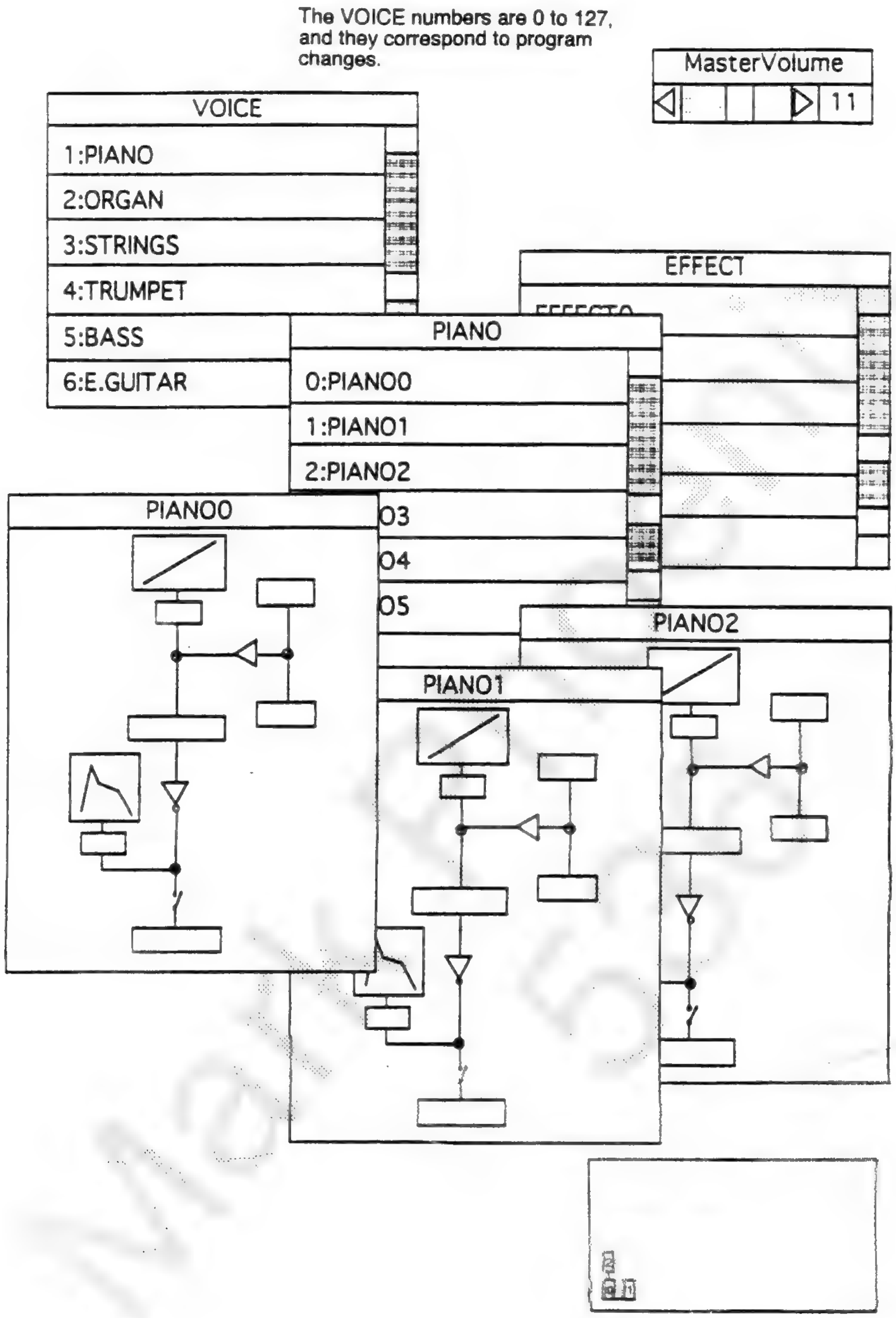
- SR: Specifies the level change per unit time. SR = SUSTAIN LEVEL/ST
- ST: Specifies the table number of the time table used for the time it takes to reach the sustain level. The time table lists the number of counts for each unit of time. To obtain the ST value, first, calculate the number of counts from the SUSTAIN TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the ST value.
  Number of counts = SUSTAIN TIME/2 (msec)
- RR: Specifies the level change per unit time.
  RR = RELEASE LEVEL/RT
- RT: Specifies the table number of the time table used for the time it takes to reach the release level. The time table lists the number of counts for each unit of time. To obtain the RT value, first, calculate the number of counts from the RELEASE TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the RT value.
  Number of counts = RELEASE TIME/2 (msec)
- Delay: Specifies the table number of the time table used for the PLFO delay time. The time table lists the number of counts for each unit of time. To obtain the Delay value, first calculate the number of counts from the PLFO DELAY TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the Delay value.
  Number of counts = PLFO DELAY TIME/2 (msec unit time)
- FRQ: Specifies the frequency per unit of time for the PLFO triangle wave.
  FRQ = DEPTH LEVEL/FRQ TIME * 2 (msec unit time)
- FDR: Specifies the fade-in amplitude change per unit of time.
  FDR = DEPTH LEVEL/FADE TIME * 2 (msec unit time)
- FDT: Specifies the table number of the time table used for the time it takes to reach the maximum fade-in level. The time table lists the number of counts for each unit of time. To obtain the FDT value, first calculate the number of counts from the PLFO FADE TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the FDT value.
  Number of counts = PLFO FADE TIME/2 (msec unit time)

3.0 Starting the Application

To start the application, double-click the sound editing tool icon from the Finder. The system first issues an inquiry to the SCSI interface to check whether the SCSP development tool is connected to the target. If the SCSP development tool is not connected, the program terminates. If the SCSP development tool is connected, the system downloads the 68000 program that is used by the sound editing tool to the target and executes it. The download is not executed if the program has already been downloaded, or if the waveform editing tool or the sound editor was previously used. The decision of whether or not to download the program is determined by the ProgramReady flag.

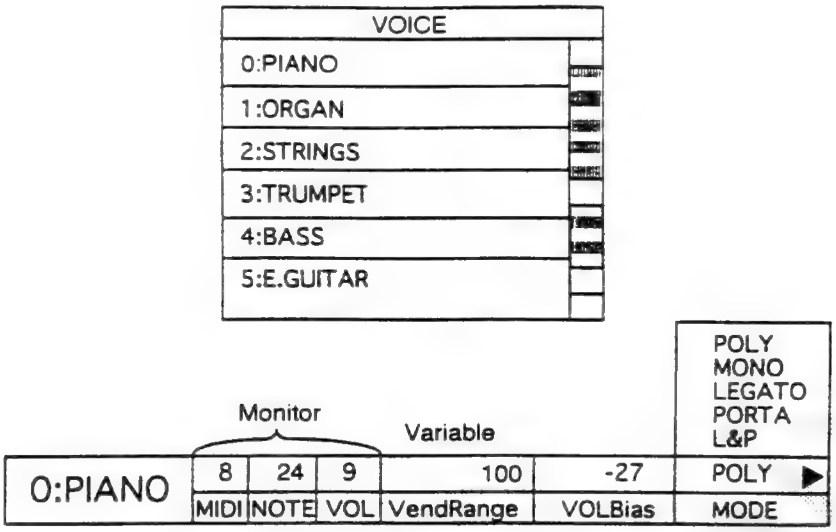
After downloading and executing the program, the system starts the sound editing tool. When the tool is started, the menu and the master volume window are displayed.

4.0 Screen Structure



The screen structure is as shown on the previous page. The basic operations of this software are performed through a menu and the following windows: Voice, Layer, Effect, Master Volume, Slot, Parameter. The use of these components are described in upcoming sections. (This document does not describe standard Macintosh operations, such as moving windows.) The Slot window, the FM Connection window, and the Parameter window can only display one sound at a time. For example, if another sound is opened through the Voice window while any of these three windows are open, all windows for that sound will be closed and windows for the new sound will be opened in its place.

Voice Window



The structure of the Voice window is as shown in the above figure. The title bar shows "VOICE," and the window contains the sound name (up to 15 bytes), a MIDI monitor, the bend range, and the play mode parameters. The Voice window allows the Voice names and the parameters to be changed. To alter the bend range, change the value directly in the parameter value display area. The allowed values are 0 to 2^{14}. To change the play mode, select a mode from the menu. The mode selections are polyphonic mode (POLY), mono mode (MONO), legato mode (LEGATO), portamento mode (PORTA), and legato and portamento mode (L&P). When legato mode, portamento mode, or legato and portamento mode is selected, the play mode switches automatically to mono mode. (Bit 0 of the play mode is 1.)

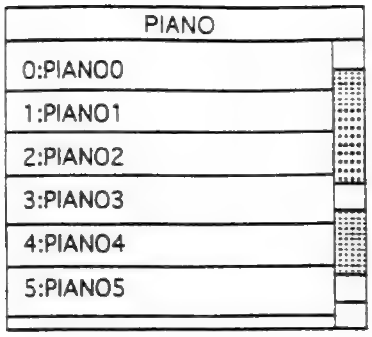
The Voice window functions as a data monitor. It tells which Voice is currently allocated to a sound, and its volume level. This information can be monitored by checking the Note, Vol, and Program of each channel in the MidiData memory address of the target. Thus, this window retrieves and displays data in real-time under any conditions.

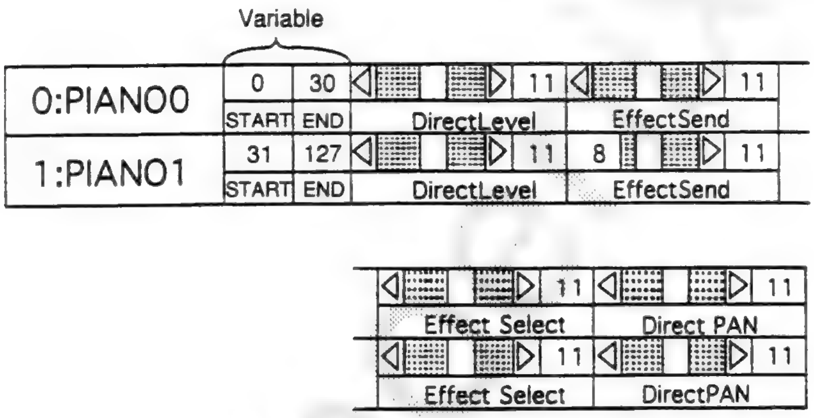
Also, double-clicking the Voice name opens the Layer window. Closing this window closes all windows except the Master Volume window.

- MIDI: Shows which MIDI channel is being used and the sound played on it.
- NOTE: Shows which note is being played.
- VOL: Shows the volume at which the note is being played.
- BendRange: Variable width setting of the bend range (0 to $[2^{14} - 1]$)
- VOLBias: Volume bias setting (-128 to 127)

When double-clicking the Voice name to open the Layer window while the number of Layers is zero, a Layer input window opens. (See **Layer Menu** on page 33.)

Layer Window



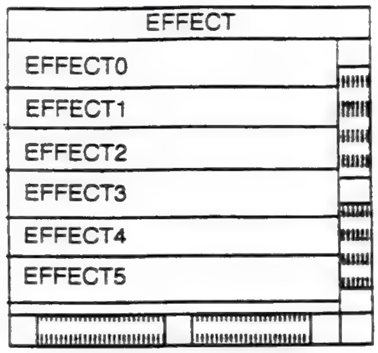


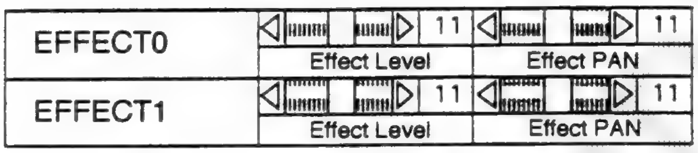
The structure of the Layer window is as shown in the above figure. The title bar shows the Voice name, and the window contains the Layer name (up to 15 bytes); the direct send (DISDL), effect send (IMXL), DSP input channel (ISEL), PAN (DIPAN) parameters, and a control bar for each parameter. The Layer window allows Layer names and parameters to be changed. To change a parameter, drag the scroll bar, click the left or right button, or enter a value directly in the parameter data display area.

Double-clicking a Layer name closes the Slot window and all related parameter windows that were opened when the previous Layer name was double-clicked. Closing the Layer window closes the Slot window and all related parameter windows.

- START: Start note setting (0 to 127)
- END: End note setting (0 to 127)
- DirectSend: DISDL setting (0 to 7)
- EffectSend: IMXL setting (0 to 7)
- Effect: ISEL setting (0 to 15)
- PAN: DIPAN setting (0 to 31)

Effect Window

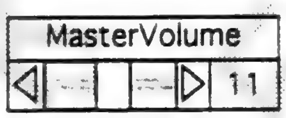




The structure of the Effect window is as shown in the above figure. The title bar shows "EFFECT," and the window contains the send return (EFSDL) and pan (EFPAN) parameters for each effect channel and a control bar for each parameter. The Effect window allows each parameter to be changed. To change a parameter, drag the scroll bar, click the left or right button, or enter a value directly in the parameter data display area.

- SendReturn: EFSDL setting (0 to 7)
- PAN: EFPAN setting (0 to 31)

Master Volume Window

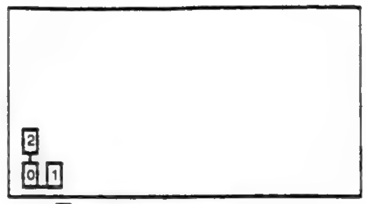


The title bar shows "MasterVolume," and the window contains a control bar for the master volume. To change the master volume, drag the scroll bar, click the left or right button, or enter a value directly in the parameter data display area.

This window is always displayed at the top of each window. The master volume directly overwrites MVOL in the SCSP. The address is the lower three bits of 100401.

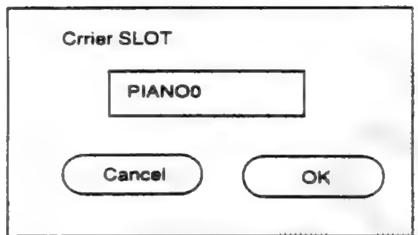
- MasterVolume: MVOL setting (0 to 15)

FM Connection Window



Tentative version

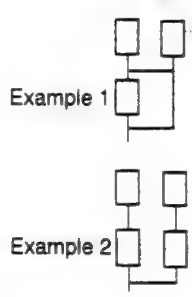
The FM Connection window displays the FM connections of the sound that is currently displayed. Initially, nothing is displayed. When you double-click the mouse, a window opens.



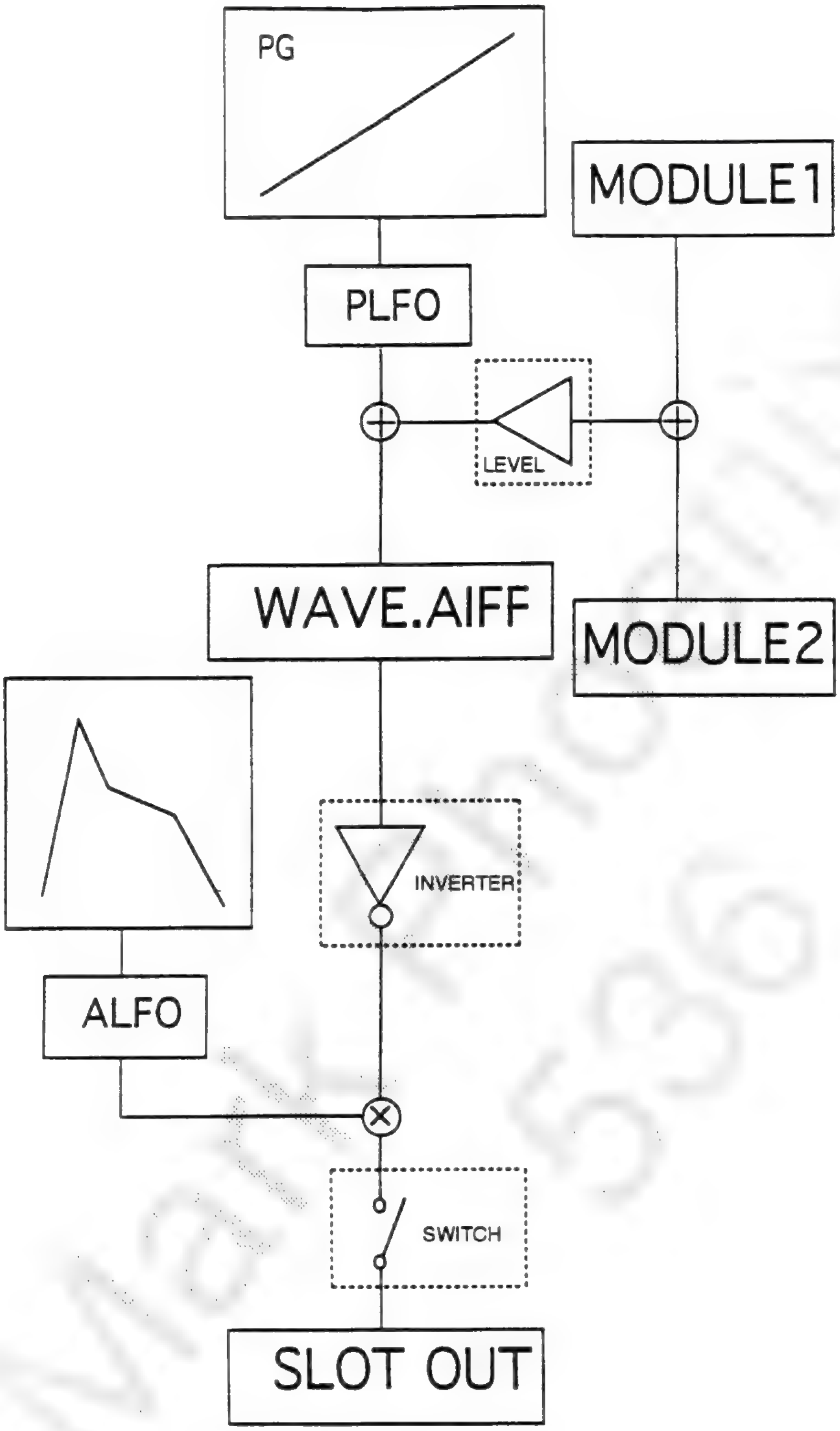
Tentative version

Select the base Slot in the window. The window has a menu where you specify the Layer name. The display shows the Layers in the Voice which have the same start and end notes as the base Layer. First, the base Slot is displayed as a box □. Next, if the Slot in the Slot window has Layers, those Layers are written above the base Slot. In this revision, two Layers can be displayed above the base Slot. The upper Layer X is displayed directly above the base Slot. The lower Layer Y is displayed diagonally above the base Slot. If X and Y are the same Layer, only one is displayed. All of the Layers are displayed in this way up to the top Layer. A Layer with the same start and end notes which is not yet displayed will not be displayed. This Layer is recognized as one of the Layers of a multilayer.

Note: With this display method, a Slot functioning as a modulator in a FM connection Layer that is set up to produce sounds will a cause an abnormal display. A configuration like the one in Example 1 cannot be displayed. In Example 2, the FM Layers are actually multilayers for a two-Slot FM connection, and only one of the Layers can be displayed.



Slot Window


PG
MODULE1
PLFO
LEVEL
WAVE.AIFF
MODULE2
INVERTER
ALFO
SWITCH
SLOT OUT

To open the Slot window, double-click a Layer name in the Layer window. The title of the Slot window is the Layer name. When this window is closed, all parameter and Slot windows related to the sound are closed. Closing a Slot window other than SLOT0 closes only that window.

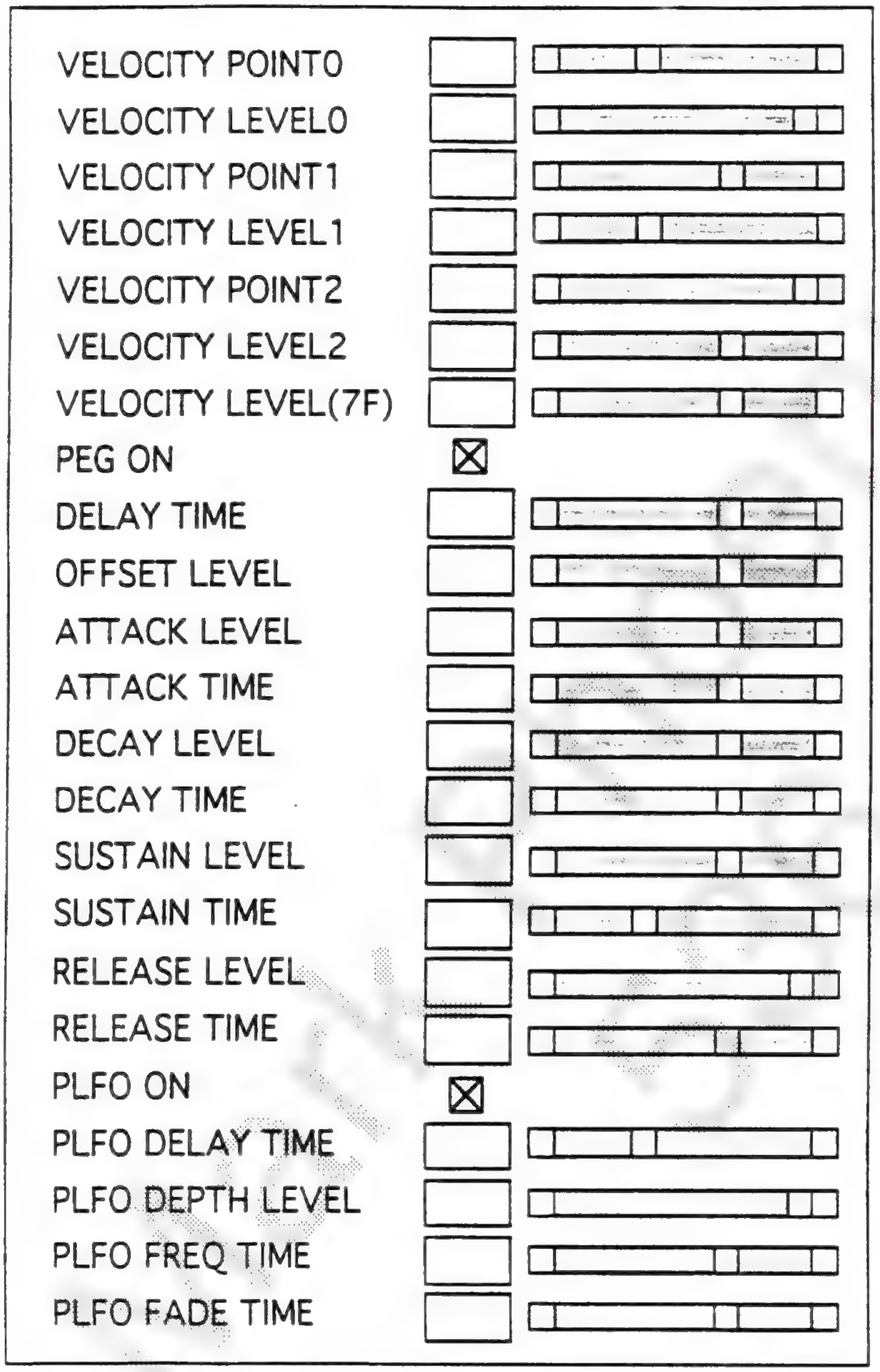
The Slot window allows the SCSP register parameters to be edited, and the Parameter window to be opened. The figure in the window represents the hardware algorithm in the SCSP. To change a parameter, click the appropriate block. To open the Parameter window, double-click the block.

Change the following parameter by clicking the appropriate block:

- Switch: OFF sets EFSDL and DISDL to 0 and simultaneously sets bit 15 of registers 16 and 17 to 1.

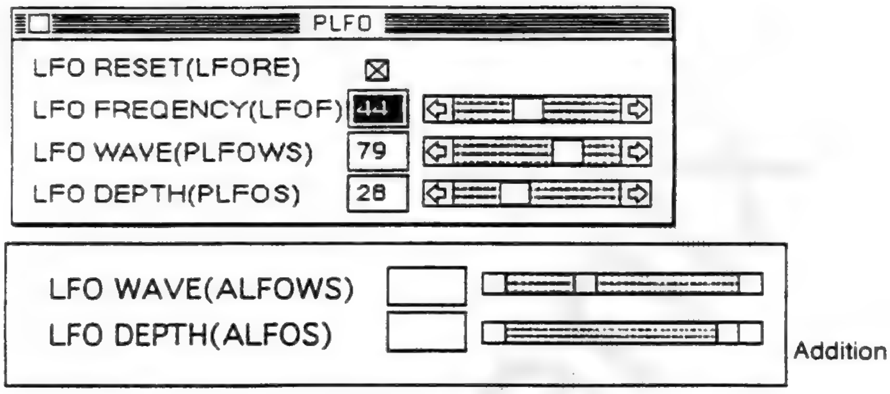
Parameter Window

The Parameter window displays the names of all of the parameters, data, and scroll bars. To change a parameter, drag the scroll bar, click the left or right button, or directly change the value in the parameter data display area.



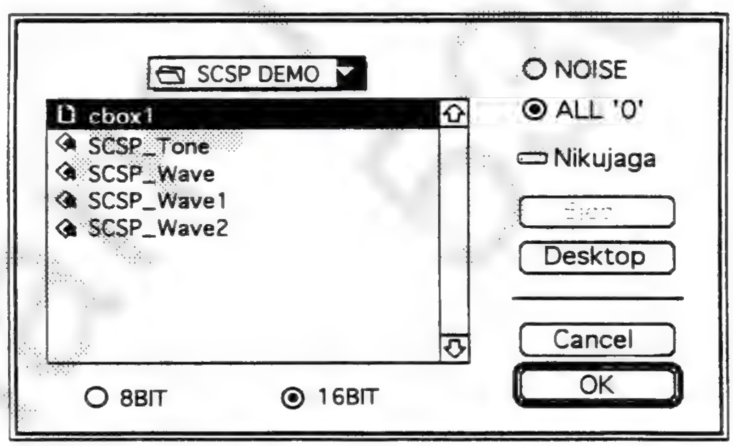
Tentative version

Double-clicking the PG block in the Slot window produces a window titled PEG, which is simular to the one shown on the previous page. The PEG window allows the pitch EG and pitch LFO parameters to be edited. The range for each LEVEL parameter is 0 to 127. The range for a TIME parameter depends on the time table. The format for the TIME parameters is as described earlier. When a value is entered and Return is pressed, the system looks for the nearest table value and displays that value. When the scroll bar is used, the system will display the table value of the next table number.



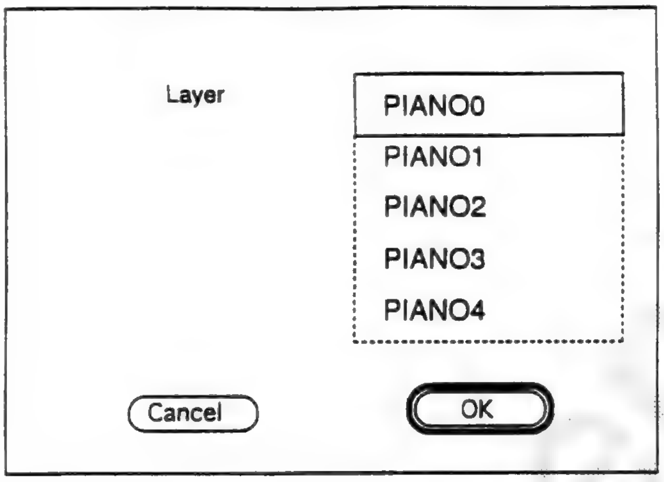
Double-clicking the PLFO block in the Slot window displays a window like the one above. This window allows the pitch LFO parameters to be edited. This window is the same as for the ALFO block.

- LFOF: 0 to 31
- PLFOWS: 0 to 3
- PLFOS: 0 to 7
- ALFOWS: 0 to 3
- ALFOS: 0 to 7



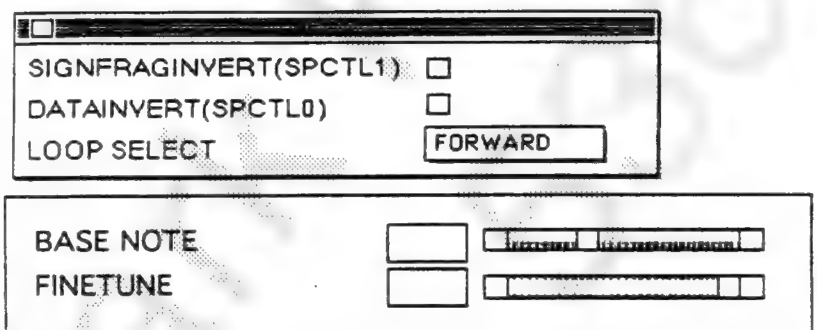
Add Layer to the radio buttons.

Double-clicking the WAVE.AIFFO block in the Slot window produces a window like the one on the previous page. This window allows the SA, LSA, LEA, SSCTL, and 8B parameters to be copied. LSA and LEA are read from AIFF, and the other parameters are edited with the radio buttons. Select OK in this window to overwrite the format. For instructions on overwriting the format, refer to the menu items. After selecting the Layer radio button, the following window appears:



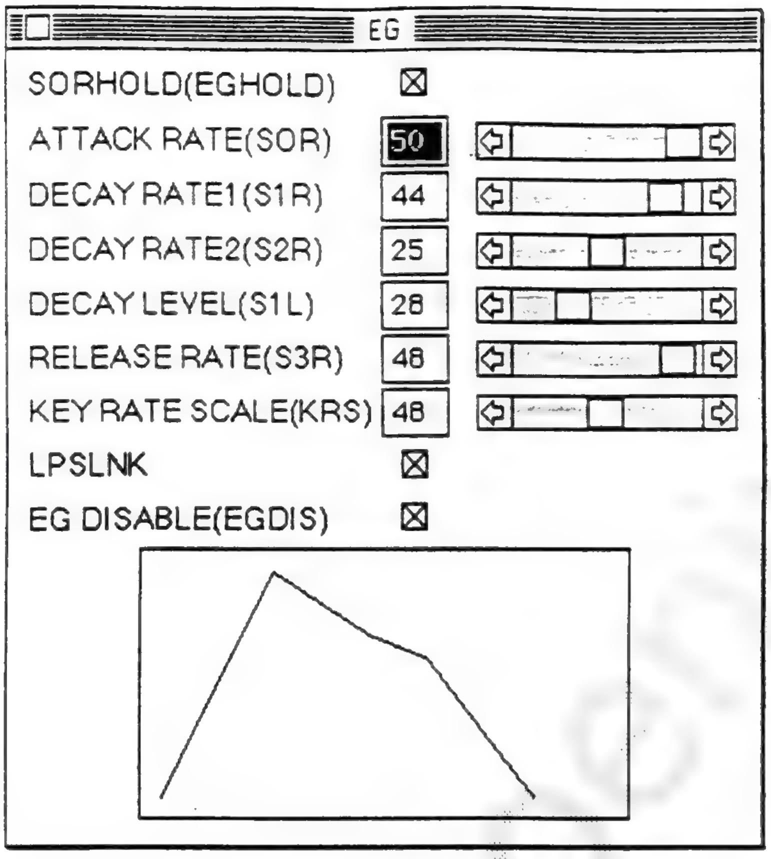
From this window, it is possible to select not only a Layer of the Voice being edited, but also one of the Layers of another Voice. When a Layer is selected, the values for the SA, LSA, LEA, SSCTL, and 8B parameters become the same as the values in the selected Layer.

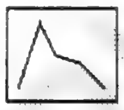
- 8BIT and 16BIT: 8BIT sets the 8B parameter to 1.
- NOISE and ALL'0': NOISE sets SSCTL to 1. ALL'0' sets SSCTL to 2.



Add to bottom.

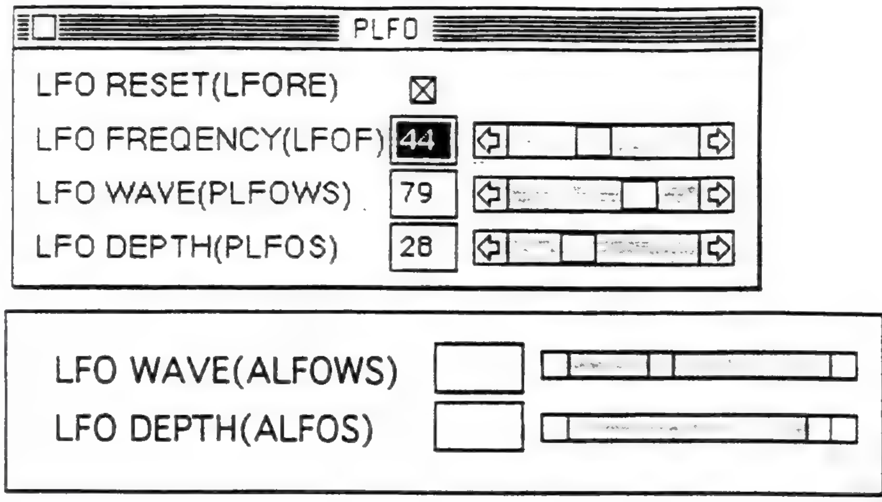
Double-clicking the Inverter block in the Slot window displays a window like the one above. This window allows the editing of LPCTL, SPCTL, and the BASE NOTE and FINETUNE parameters of the Layer. The range for BASE NOTE and FINETUNE is 0 to 127. When the parameter is checked off, the corresponding register is set to 1. The LOOP SELECT parameter has four settings: FORWARD, REVERSE, ALTERNATE, and OFF.



After double-clicking  in the Slot window, a window like the one above appears. This window allows the EG parameters to be edited. The EG envelope is displayed only in this window at the bottom, and the envelope changes when the parameters are edited. You can also edit the parameters by dragging the envelope.

- ATTACK RATE: AR setting (0 to 31)
- DECAY RATE1: D1R setting (0 to 31)
- DECAY RATE2: D2R setting (0 to 31)
- DECAY LEVEL: DL setting (0 to 31)
- RELEASE RATE: RR setting (0 to 31)
- KEY RATE SCALE: KRS setting (0 to 15)

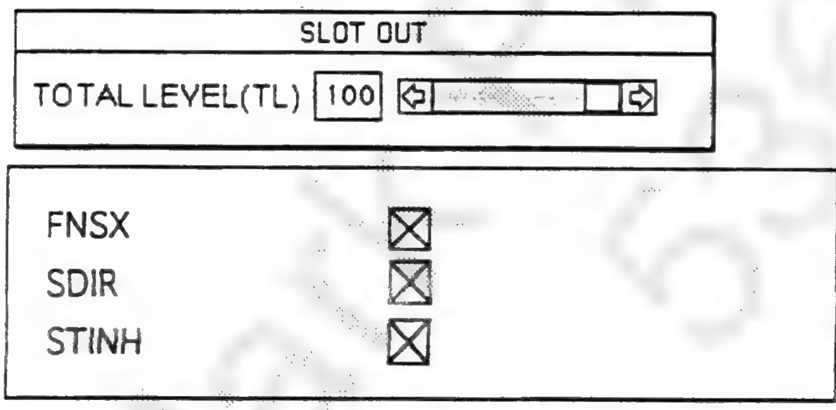
Checking off a parameter sets the corresponding bit to 1.



Addition

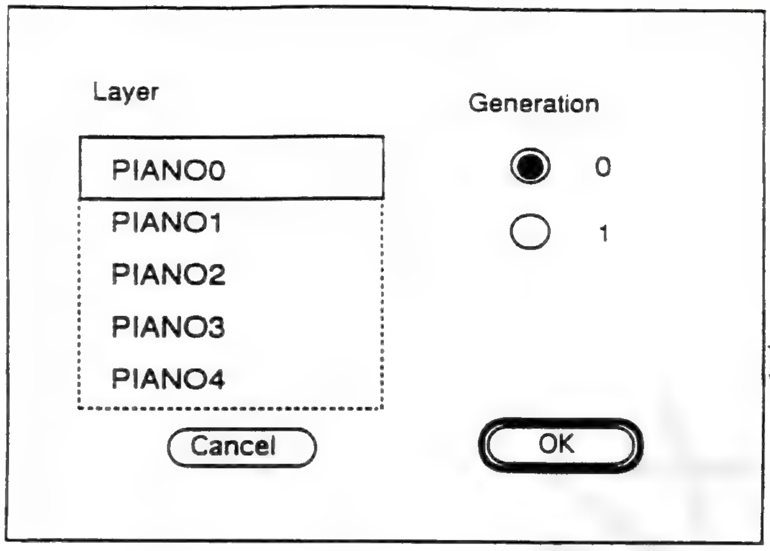
Double-clicking the ALFO block in the Slot window displays a window like the one above. This window allows the pitch LFO parameters to be edited. This window is the same as for the PLFO block.

- LFOF: 0 to 31
- PLFOWS: 0 to 3
- PLFOS: 0 to 7
- ALFOWS: 0 to 3
- ALFOS: 0 to 7



Addition

Double-clicking SLOTOUT in the Slot window produces a window like the one above. This window allows TL, FNSX, SDIR, and STINH to be edited. The range for these parameters is 0 to 127.

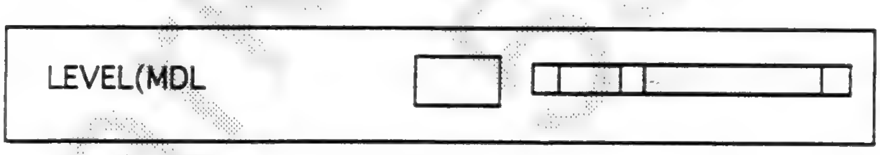


Tentative version

Double-clicking MODULE1 or MODULE2 in the Slot window produces a window like the one above. The Layer side of the window lists the Layer names. The list is a menu from which a Layer name is selected. This window allows MDXSL (the upper Slot in the Slot window) and MDYSL (the lower Slot in the Slot window) to be changed.

This window also allows the generation to be selected with the radio button. To select the latest data, click 0. To select the data of the previous generation, click 1. The data in bit 7 of MDXSL and MDYSL is changed. Selecting 0 sets bit 7 to 0; selecting 1 sets bit 7 to 1.

The parameters that are set in this window are not the actual Slots that are sounded by the SCSP. This is because the 68000 program is a DVA and can assign this sound to any Slot.



Tentative version

Double-clicking  in the Slot window displays a window like the one above. This window allows MDL to be edited. The range for this parameter is 0 to 15.

5.0 Menus

File Menu

New
Creates a new sound-format file. When this option is selected, the number of Voices (see Voice... in the Edit menu) must be specified. The Voice window and the Mixer window are both displayed. The format that is created with this function has a Voice Chunk that accommodates only the specified number of Voices. This data is transferred to the application memory and is simultaneously downloaded to the address that is stored in the Voice offset address of the target.

Open...
Loads the sound format that is stored in the file. Only SCSP formats can be loaded. This data is transferred to the application memory and is simultaneously downloaded to the address that is stored in the Voice offset address of the target.

Save ...
Saves the edited sound format to a SCSP format file. If the sound format that is being saved is data that was loaded from a file, the format is written over the same file. If the sound format is being edited for the first time, this function operates in the same manner as the *Save As...* function.

Save As...
Saves the edited sound format to a different file name or in SCSPBIN format. Radio buttons allow either the SCSP or SCSPBIN format to be selected. To save the data to a file, enter the file name and click the Save button.

Close
Closes the active window. This function also automatically closes the windows in Layers under the active window. Closing the Slot window also closes all parameter windows. Closing the Layer window closes the Slot window. Closing the Voice window closes the Layer window. If the active window is the Voice window and data has been edited but not saved, a prompt will give you the option to save the data.

Voice Info
Displays the total number of bytes in the format being edited. The total number of bytes is the number of bytes in the SCSPBIN format minus the first 12 bytes.

Quit
Terminates the application. If QUIT is selected before edited data has been saved, a prompt for saving the data will appear.

Edit Menu

Cut

This function is disabled in the Parameter and Slot windows. When used in the Layer window, this function saves all of the information in the Layer Chunk being edited to the clipboard. At the same time, it deletes the Layer from the format while formatting it. When the format is optimized, the Voice Chunk, the number of Layers, and the Layer numbers are modified.

When used in the Voice window, this function saves all of the information in the Voice Chunk being edited and the information in the accompanying Layers to the clipboard. At the same time, it deletes the Voice and the Layers from the format while optimizing it. When the format is optimized, the Voice Chunk is not deleted. Instead, the Voice is changed to one in which the number of Layers is 0.

Copy

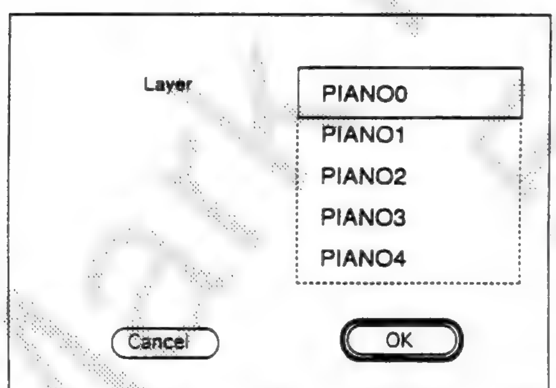
This function is disabled in the Parameter and Slot windows. When used in the Layer window, this function saves all of the information in the Layer Chunk being edited to the clipboard.

When used in the Voice window, this function saves all of the information in the Voice Chunk being edited and the information in the accompanying Layer Chunk to the clipboard.

Paste

Pastes the clipboard contents. If the clipboard contains Layer information, the information can only be pasted into the Layer window. Voice information can only be pasted to the Voice window. After pasting the information, this function optimizes the format.

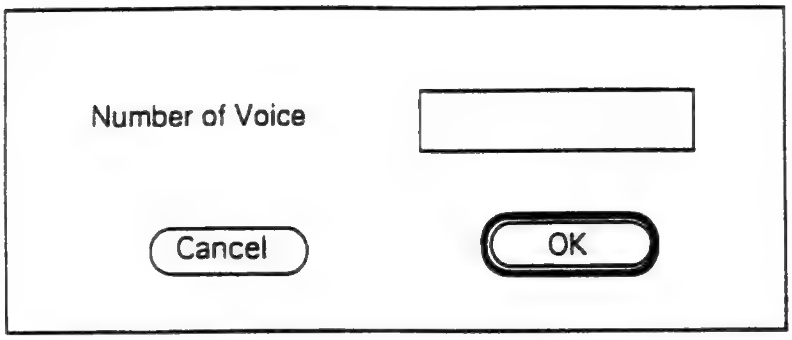
Cut Wave...



Deletes Wave data. When this function is selected, the above window appears. When a Layer is selected, this function deletes that Wave data and optimizes the format.

Voice Menu

Number...



Enables the number of Voices to be inputted. This function then optimizes the format. A value that is smaller than the current number of Voices cannot entered.

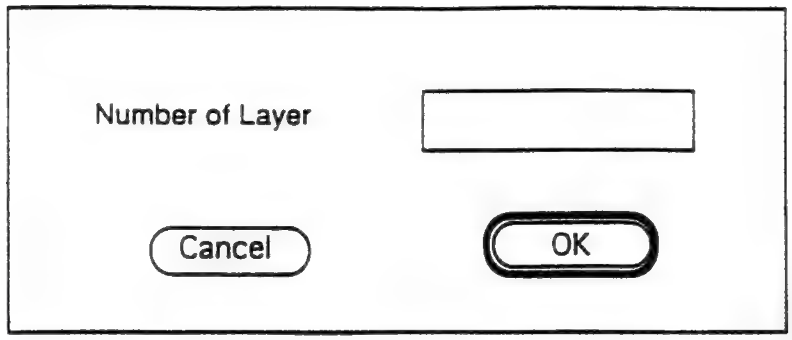
Download
Downloads the format to the address stored in the Voice offset address of the target.

Insert
Increases the number of Voices by one. This function then optimizes the format.

Delete
Decreases the number of Voices by one. This function then optimizes the format.

Layer Menu

Number...



Enables the number of Layers to be inputted. This function then optimizes the format. A value that is smaller than the current number of Layers cannot be entered.

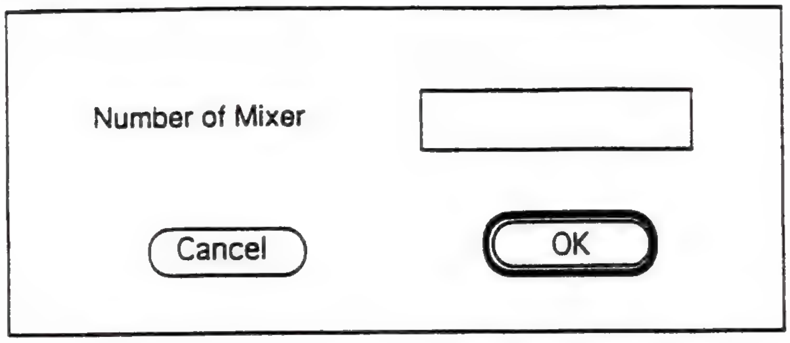
Download
Downloads the format to the address stored in the <u>Voice offset address</u> of the target.

Insert
Increases the number of Layers by one. This function then optimizes the format.

Delete
Decreases the number of Layers by one. This function then optimizes the format.

Mixer Menu

Number...



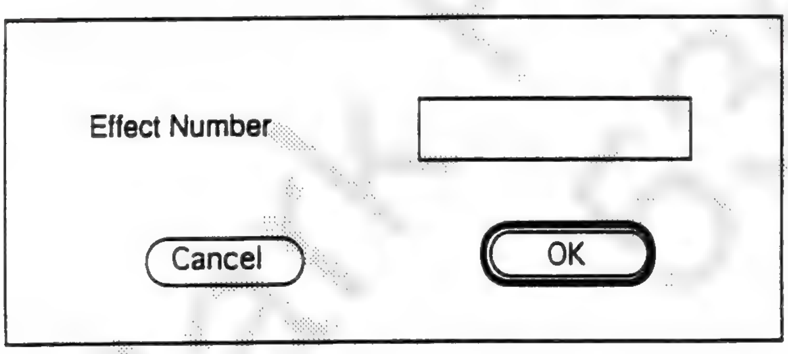
Enables the Effect number to be inputted. This function then optimizes the format. A value that is smaller than the current number of Layers cannot be entered.

Download
Downloads the format to the address stored in the Voice offset address of the target.

Insert
Increases the number of Voices by one. This function then optimizes the format.

Delete
Decreases the number of Voices by one. This function then optimizes the format.

Edit...
Choose the Effect to be edited.



After an Effect is selected, this function closes the current window being edited and opens a new window. It then switches the Effect.

6.0 Format Optimization

To maximize the efficient use of memory, this editing tool deletes the gaps in the format that are formed when Voices or Layers are cut or added, or when Wave data is edited. When the format is changed in this manner, parameters, such as start address, number of Layers, and offset, are modified. The process of changing the parameters and deleting the gaps is called Format Optimization.

Described below are the optimization methods for each edit type. (These methods apply to the SCSPBIN format.)

Cut

Voice
If Cut is executed when the Voice window is active, format optimization becomes necessary. The following parameters are modified:

- Layer offset
- Wave offset
- Layer number

Layer Offset
When a Voice is deleted, that portion of the Voice Chunk is eliminated entirely. The Layer offset is therefore reduced by the size of that portion. When a Voice with n Layers is deleted, the offset is reduced by the following:

5 + 3 * n bytes

Wave Offset
The Wave offset must be reduced by the size of the deleted Layer offset plus the size of the deleted Layers. The Wave offset is therefore reduced by the following:

(5 + 3 * n) + 47 * n bytes

Layer Number
The Voices that precede the deleted Voice do not change. However, the Layer numbers in the Voice Chunk that follow the deleted Voice must be reduced by the number of Layers that were deleted.

Layer
If Cut is executed when the Layer window is active, format optimization becomes necessary. The following parameters are modified:

- Wave offset
- Layer number

Wave Offset
The Wave offset is reduced by 47 bytes per Layer.

Layer Number
The Layer numbers after the deleted Layer number are reduced by one.

Wave
If a Wave is cut, format optimization becomes necessary. The following parameters are modified:

- Start address
- Wave number

Start address
In Wave numbers larger than the deleted Wave number, the start address must be reduced by the size of the Wave that was deleted.

Wave number
The Wave numbers that are larger than the deleted Wave number are reduced by one.

7.0 Addendum

For Voice
The following parameters are changed:

- Layer offset
- Wave offset
- Layer number

SEGA OF AMERICA, INC.
Consumer Products Division

SCSP/DSP Effect Module Specifications (Tentative)

Doc. # ST-69-121693

9/3/93, Ver. 1.00

YAMAHA CORPORATION

1.0 Introduction

The SCSP/DSP Effect modules are function-specific software modules that users can link together with the SCSP/DSP Linker software to create their own DSP microprograms.

2.0 Effect Modules Scheduled to Be Developed

1. Reverb(s)
2. Early Reflection(s)
3. Echo (Delay) (s)
4. Pitch Shifter(s)
5. Chorus
6. Flanger
7. Symphonic
8. Surround
9. Voice Cancel
10. Auto Pan
11. Phaser
12. Distortion
13. Filter
14. Parametric EQ

Notes:

1. The modules with (s) at the end of their names are scheduled to be released with subsets of the effect family. For the modulation type modules (5, 6, 7, and 11), prototype effect subsets are scheduled to be created and tested. Whether or not the subsets will be released has not been determined.

2. The values indicated in the following specifications for number of steps and Delay buffer RAM size are tentative and may be changed during the module development and evaluation stages. The values enclosed in parentheses are the subset edition values for each effect subset module.

3. The parameters type and the value ranges may be changed during the module development and evaluation stages.

Reverb(s)

Effect	Generates reverb.
Number of steps	34 (28)
Delay buffer RAM size	15 [kwords]

Parameter	Value	Remarks
Type	Hall/Room/Vocal/Plate/others	
Initial Delay	0.1 to several 10's [ms]	
Diffusion	0 to 10	
Reverb Time	0.3 to several 10's [s]	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Comments: It is possible to partially compensate for the decreases in the number of steps and Delay buffer RAM size by adjusting the quality and reverb time of the effect.

Early Reflection(s)

Effect	Generates Reverb-type effects using early reflection.
Number of steps	100 (60)
RAM size used for delay	13 [kwords]

Parameter	Value	Remarks
Type	Hall/Random/Reverse/ Spring/others	
Initial Delay	0.1 to 100[ms]	
Liveness	0 to 10	
Diffusion	0 to 10	Unavailable in the subset version.
Room Size	0.1 to 10	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Comments: The full version is stereo, and the subset version is mono. It is possible to partially compensate for decreases in number of steps and Delay buffer RAM size by adjusting the quality of the Early Reflection sound.

Echo (Delay) (s)

Effect	Generates echo sounds.
Number of steps	20 (10)
RAM size used for delay	26 (13) [kwords]

Parameter	Value	Remarks
Delay Time Left	0.1 to 300 [ms]	
Feed Back Left	-99 to +99%	
Delay Time Right	0.1 to 300 [ms]	
Feed Back Right	-99 to +99 [%]	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Comments: The full version is stereo. The subset version is mono (simulated stereo) and features simplified preprocessing.

Pitch Shifter (s)

Effect	Generates three independently pitch-shifted sounds in addition to the direct sound.
Number of steps	64 (32)
RAM size used for delay	15 (2) [kwords]

Parameter	Value	Remarks
Pitch1	-12 to +12	Related to SCSP synthesizer settings.
Fine1	-99 to +99	Related to SCSP synthesizer settings.
Delay1	0.1 to 300 [ms]	Unavailable in the subset version.
Pitch2	-12 to +12	Related to SCSP synthesizer settings. Unavailable in the subset version.
Fine2	-99 to +99	Related to SCSP synthesizer settings. Unavailable in the subset version.
Delay2	0.1 to 300 [ms]	Unavailable in the subset version.
Pitch3	-12 to +12	Related to SCSP synthesizer settings. Unavailable in the subset version.
Fine3	-99 to +99	Related to SCSP synthesizer settings. Unavailable in the subset version.
Delay3	0.1 to 300 [ms]	Unavailable in the subset version.
Feedback	-99 to 99 [%]	
Pitch1 Level	0 to 100%	
Pitch2 Level	0 to 100%	Unavailable in the subset version.
Pitch3 Level	0 to 100%	Unavailable in the subset version.
Direct Level	0 to 100%	

Comments: The subset version generates only one pitch-shifted sound, and does not support control over delay.

Chorus

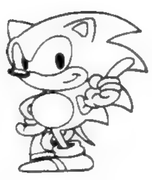
Effect	Generates a chorus effect.
Number of steps	22
RAM size used for delay	1 [kword]

Parameter	Value	Remarks
Rate	Arbitrary	Related to SCSP synthesizer settings.
Amp Depth	0 to 100 [%]	
Pitch Depth	0 to 100 [%]	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Flanger

Effect	Generates a flanging effect.
Number of steps	20
RAM size used for delay	2 [kwords]

Parameter	Value	Remarks
Rate	Arbitrary	Related to SCSP synthesizer settings.
Mod Delay	0.1 to 20 [ms]	
Feedback	-99 to 99 [%]	
Depth	0 to 100 [%]	
Effect Level	0 to 100%	
Direct Level	0 to 100%	



Symphonic

Effect	Generates complex chorus-type pitch changes.
Number of steps	21
RAM size used for delay	1 [kword]

Parameter	Value	Remarks
Rate	Arbitrary	Related to SCSP synthesizer settings.
Depth	0 to 100 [%]	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Surround

Effect	Generates reverb.
Number of steps	23
RAM size used for delay	15 [kwords]

Parameter	Value	Remarks
Liveness	0 to 10	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Voice Cancel

Effect	Reduces the volume level of the vocal band that is normally positioned in the center pan position.
Number of steps	36
RAM size used for delay	0 [kword]

Parameter	Value	Remarks
Effect	ON/OFF	

Auto Pan

Effect	Moves sound image to the left/right.
Number of steps	4
RAM size used for delay	0 [kword]

Parameter	Value	Remarks
Rate	Arbitrary	Can be set to any value by the sound CPU program.
Depth	Arbitrary	Can be set to any value by the sound CPU program.

Phaser

Effect	Generates rotary speaker-type effect.
Number of steps	22
RAM size used for delay	2 [kwords]

Parameter	Value	Remarks
Rate	Arbitrary	Related to SCSP synthesizer settings.
Depth	0 to 100 [%]	
Mod Delay	0.1 to 20 [ms]	
Effect Level	0 to 100%	
Direct Level	0 to 100%	

Distortion

Effect	Distorts the original sound.
Number of steps	20
RAM size used for delay	0 [kwords]

Parameter	Value	Remarks
Distortion	0 to 100 [%]	
Output Level	0 to 100 [%]	

Filter

Effect	Cuts the specified frequency band.
Number of steps	5
RAM size used for delay	0 [kword]

Parameter	Value	Remarks
Type	Low Pass/High Pass/Band Pass	
Frequency	About 30 [Hz] to 15 [KHz]	

Comments: This module can be used as a dynamic filter by dynamically rewriting the filter coefficients with the sound CPU.

Parametric EQ

Effect	Boosts or cuts frequencies around the specified frequency.
Number of steps	5
RAM size used for delay	0 [kword]

Parameter	Value	Remarks
Frequency	About 30 [Hz] to 15 [KHz]	
Gain	-12 to +12 [dB]	
Q	Low/High	

Comments: This module can be used to produce wow wow effects by dynamically rewriting the filter coefficients with the sound CPU.

SEGA OF AMERICA, INC.
Consumer Products Division

SCSP/DSP Linker Specifications

Doc. # ST-70-121693

Note:	This document is a tentative version that is based on the Tool Development Specifications v1.00, which was issued on August 18, 1993. Please note that the specifications described in this document may be changed or corrected at any time.

SCSP/DSP Linker Specifications

1.0 Main Function

The SCSP/DSP Linker links function-specific microprogram fragments (called "modules") and consolidates them into one program.

2.0 Functional Overview

The Effect Algorithm Edit screen allows the arranging of objects that represent Effect modules and I/O registers, and connecting them with lines to set the signal flow (see Figures 1 and 2).

Each object in the Edit screen has an area on the left or right edge that is designated as the input or output port, respectively. Connect these areas with click and drag operations. To assign the DSP input and output, arrange the I/O module objects in the Edit screen and connect them to the Effect module object.

To edit the parameters in an Effect or I/O module object, double click the object. A Module Edit window for parameter editing opens (see Figure 3). Several objects can be opened and edited at the same time.

The effect algoritm modules are linked together in the manner described above and then downloaded into the target.

3.0 Menus

File Menu

New
Opens an empty Effect Algorithm Edit screen. This screen has the layout of a standard window, and it accepts operations such as window size changes and scrolling.

Open...
Prompts for the selection of a file name and then opens the Effect Algorithm Edit screen.

Save...
Saves the effect algorithm that is currently being edited.

SaveAs...
Changes the name of the effect algorithm that is currently being edited and saves the algorithm. This function opens a standard file dialog box in which the filename is entered and then saved.

Close
Closes the Effect Algorithm Edit window that is currently being edited. This function displays a warning if the data has been changed but not saved.

Quit
Quits the Linker.

Edit Menu

Undo
Cancels the previous edit.

Cut
Deletes the selected object or connection line. If an object is deleted, the lines that are connected to that object are also deleted. The deleted object and its parameters are placed in the Clipboard.

Copy
The selected object and its parameters are copied and placed in the Clipboard.

Paste
Places the object that was placed in the Clipboard by Cut or Copy to an appropriate position in the Edit screen. If the position is incorrect, drag the object to correct the position.

SelectAll
Selects all objects and connections.

AlterToGlobal
Replaces the selected coefficient value with a globally defined value.

RevertToLocal
Restores the selected coefficient value to its local value.

Window Menu

EffectModules

Opens the Effect Module Selection window. The module objects are arranged in the selection window as icons on the desktop. Select the module object that to use, then position the object by dragging and dropping it in the Algorithm Edit window.

I/O Modules

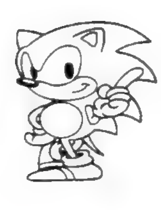
Opens the I/O Module Selection window. Position the module objects in the Algorithm Edit window with the same operation used for Effect modules.

Parameters

Opens a window for editing the parameters in the selected Effect or I/O module (see Figure 3). This operation can also be executed by double-clicking a module object.

SizeToFit

Reduces the size of the Effect Algorithm Edit screen contents so that the image of the entire algorithm can be viewed. In this state, algorithm editing is disabled. If the entire algorithm fits in the normal Edit screen, this function does not change the size of the contents. Returns to the normal edit-enabled screen when this menu item is selected again.



Process Menu

Link
Checks whether the hardware specification limits, such as the number of program steps in the current effect structure and the internal memory usage size, have been exceeded. It then displays the results and executes the link operation if linking is possible.

Download
Downloads the linked program to the SCSP.

4.0 #Module Types

DSP Input Module

The DSP input module reads data from either the 16 input buffers, #0 to #15, in the DSP unit or from the two external extension input buffers , #0 and #1. It then multiplies the data with a user-specified coefficient (send level) and writes the result to the output port.

The following data items can be edited:

- Data read destination (DSP input buffers #0 to #15)
- Send level

DSP Output Module

The DSP output module reads data from the input port, multiplies the data with a user-specified coefficient (return level), and writes the result to DSP output buffers #0 to #15.

The following data items can be edited:

- Return level
- Data write destination (DSP output buffers #0 to #15)

Effect Module

The Effect module performs effects processing on the input port data and outputs the result to the output port.

The following data items can be edited:

- Effect parameters (The Linker provides a separate editor for each Effect module.)
- Input buffer specifications for modulation purposes.

5.0 Algorithm Editing

1) **Positioning Modules**
Retrieve any module(s) from the Window Menu's Effect (or I/O) and position them in the Algorithm Edit Window.

2) **Connecting Modules**
Using the connection tool (found in the tool box), connect the input and output ports of the module objects that were placed in the Edit window. Assign module ID numbers to the modules in the order they are connected. Each connection connects two modules. If both of the modules do not have module ID numbers assigned to them, the smaller module number is set on the module where the output connection is made (output side). The ID numbers are necessary for the Linker to link the modules properly. If a module's input and output ports are disconnected, the ID number setting of that module is deleted and each of the following ID numbers are reduced by one. The module ID numbers can also be edited manually. The ID numbers must be continuous and must not be duplicated. Each time an ID number is changed, the Linker adjusts the ID numbers accordingly.

For example, if a signal that is processed in Module A is used as an input source for Module B, then the ID number of A must be smaller than the ID number of B. Otherwise, the program may not operate properly.

3) **Defining/Using Global Coefficients**
The locally defined coefficients in a module are usually assigned a coefficient RAM of one word per coefficient. When a coefficient is defined as a global coefficient, two or more modules can share a one-word coefficient.

To define a global coefficient, open the Global Coefficient window, and specify the object name of the global coefficient and its value (see Figure 4). Select and delete any number of coefficients from the global coefficient list in the window.

To replace a local coefficient with a global coefficient, open the Module Parameter Edit screen, select the value of the local coefficient to be replaced, and then select Edit-AlterToGlobal from the menu. A list of the currently defined global coefficients and their values (same as the contents of the Global Coefficient window) are displayed. From this list, select and edit the desired global coefficient. When the operation is completed, the name (character string) of the globally defined coefficient is displayed in the Module Parameter Edit screen's local coefficient value display. (See Figure 5.)

4) Specifying the DSP Modulator Input Buffer

In order to input modulator signals from a sound source into a module, specify the DSP input buffer that will receive the signals. This operation sets up an X(M) button in the module object (GUI) of the module that will input the modulator signals from the sound source. Clicking the button opens a dialog box for specifying the DSP input buffer. A DSP input buffer must be specified.

Normally a one-word DSP input buffer is assigned to one modulator group. However, if the same DSP input buffer is specified in two or more modules, a one-word DSP modulator input buffer may be shared.

6.0 Link Result Data File (tentative name)

The link result data is output to a text file. The data includes user-specified parameters, DSP input and output buffers assigned to input and output modules, DSP modulator input buffer assignment, and addresses of the coefficient RAM and address constant RAM that were automatically assigned by the Linker. Use this file when writing programs for the CPU that controls the SCSP/DSP.

7.0 Module Editor

Double-clicking a module object opens the Parameter Edit screen. This screen allows the module parameters to be edited similar to a commercial signal processor, without having to directly modify the DSP's internal coefficients or addresses. The parameters that can be edited include:

- Delay time → time [ms]
- Mix level → percentage [%]
- Filter cutoff → frequency [Hz]

In order to set up this system, software routines (module editor) are needed that convert units into binary data for each module.. For details, see the Effect module specifications.

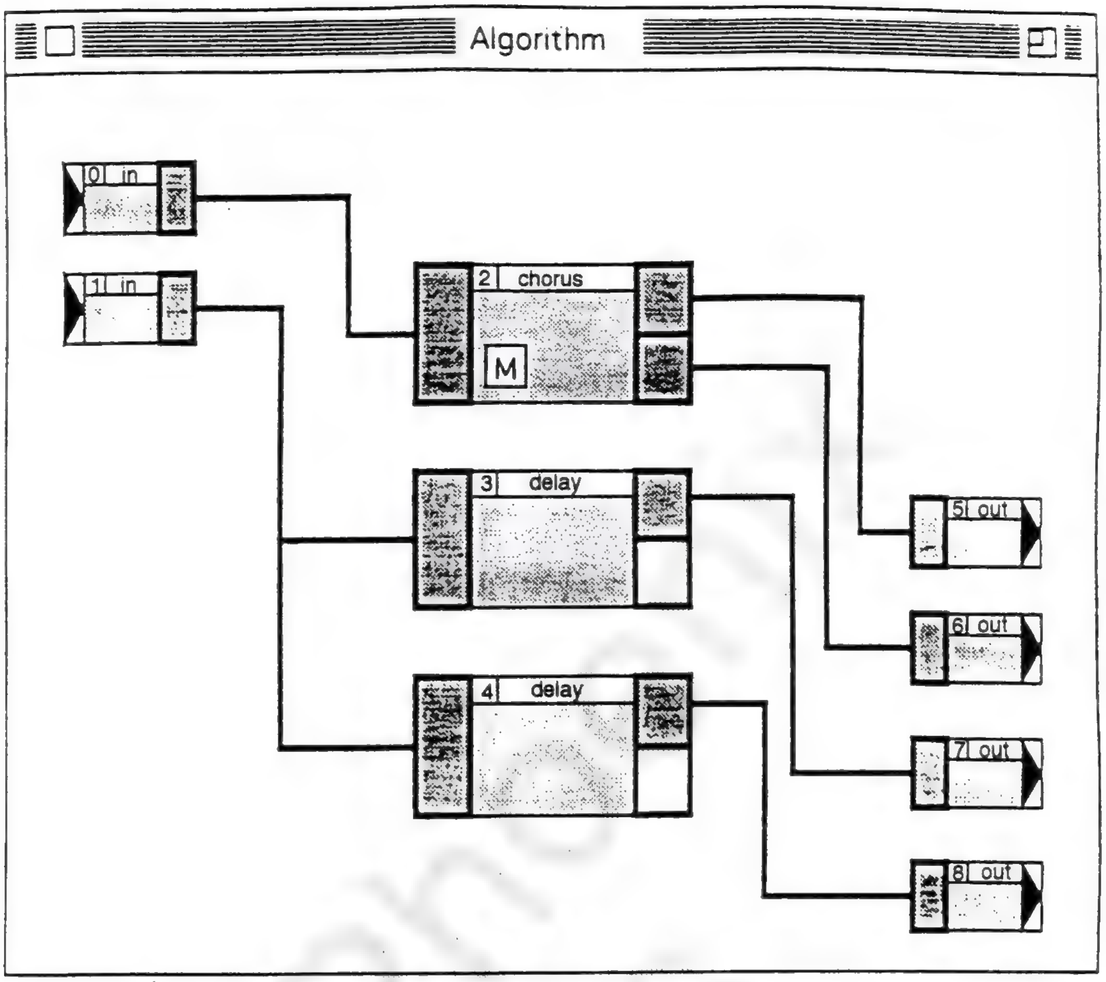


Figure 1 Effect Algorithm Edit screen

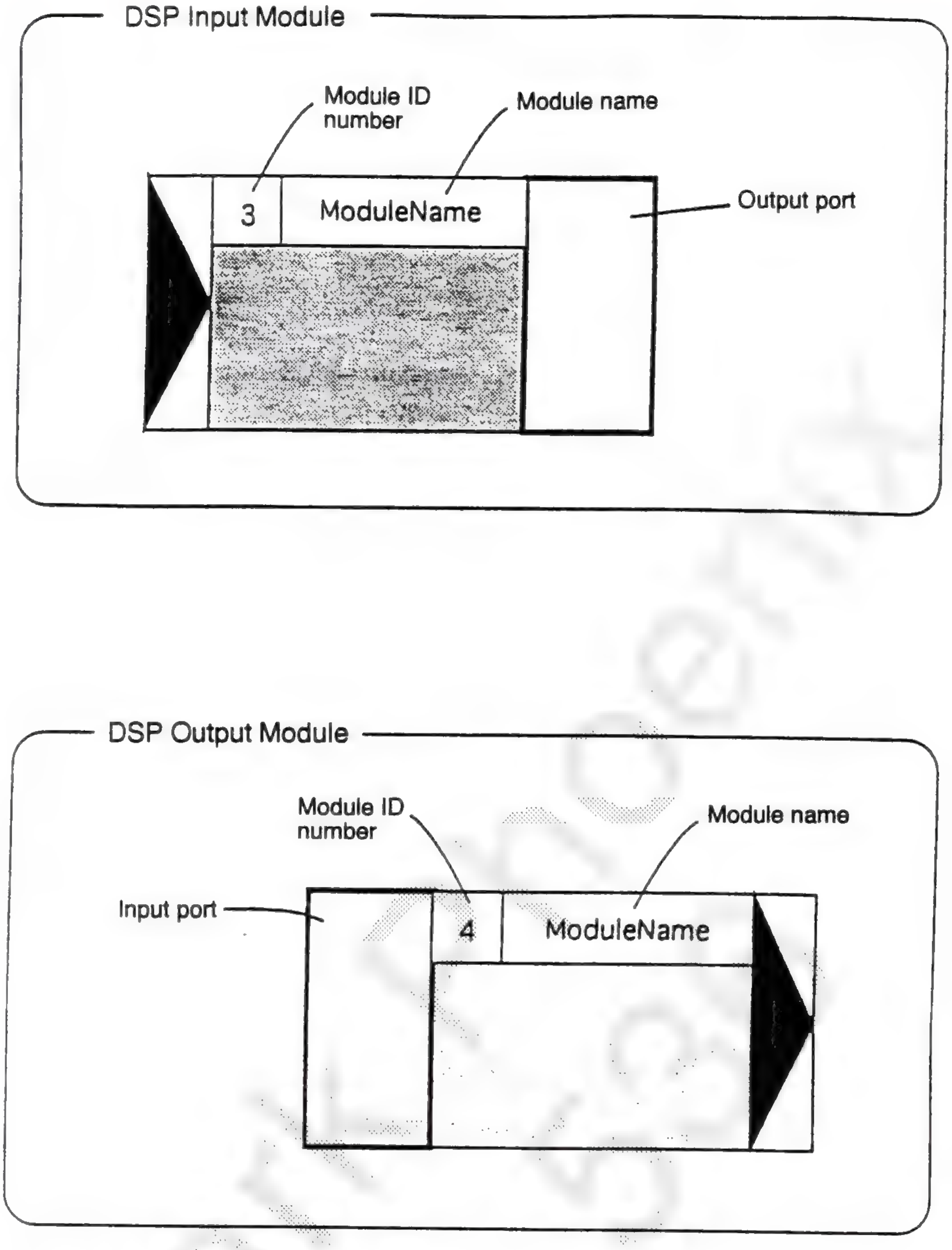


Figure 2a Module Object GUI screen (1)

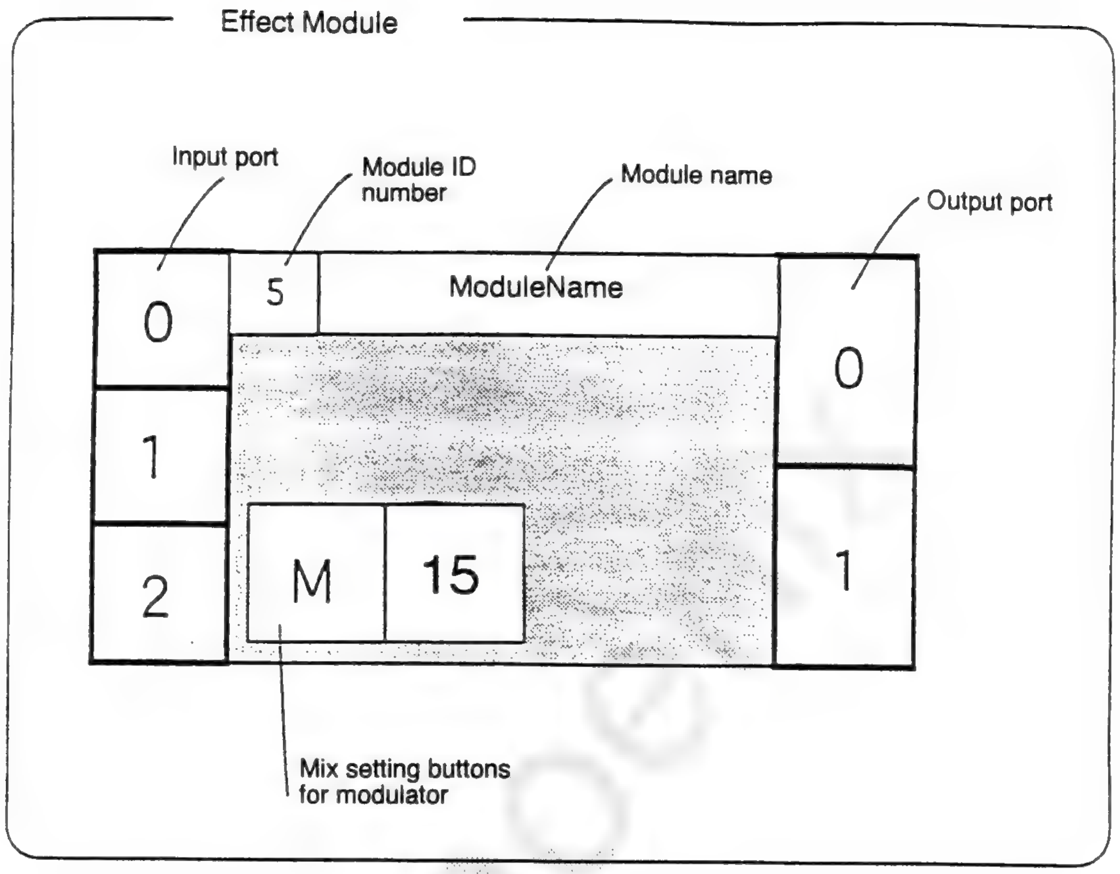


Figure 2b **Module Object GUI screen (2)**

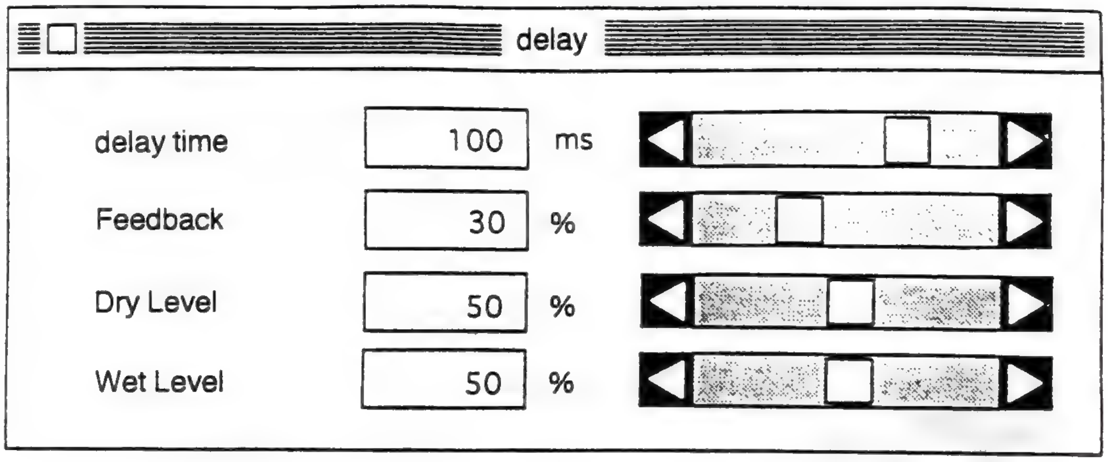


Figure 3 **Module Edit screen**

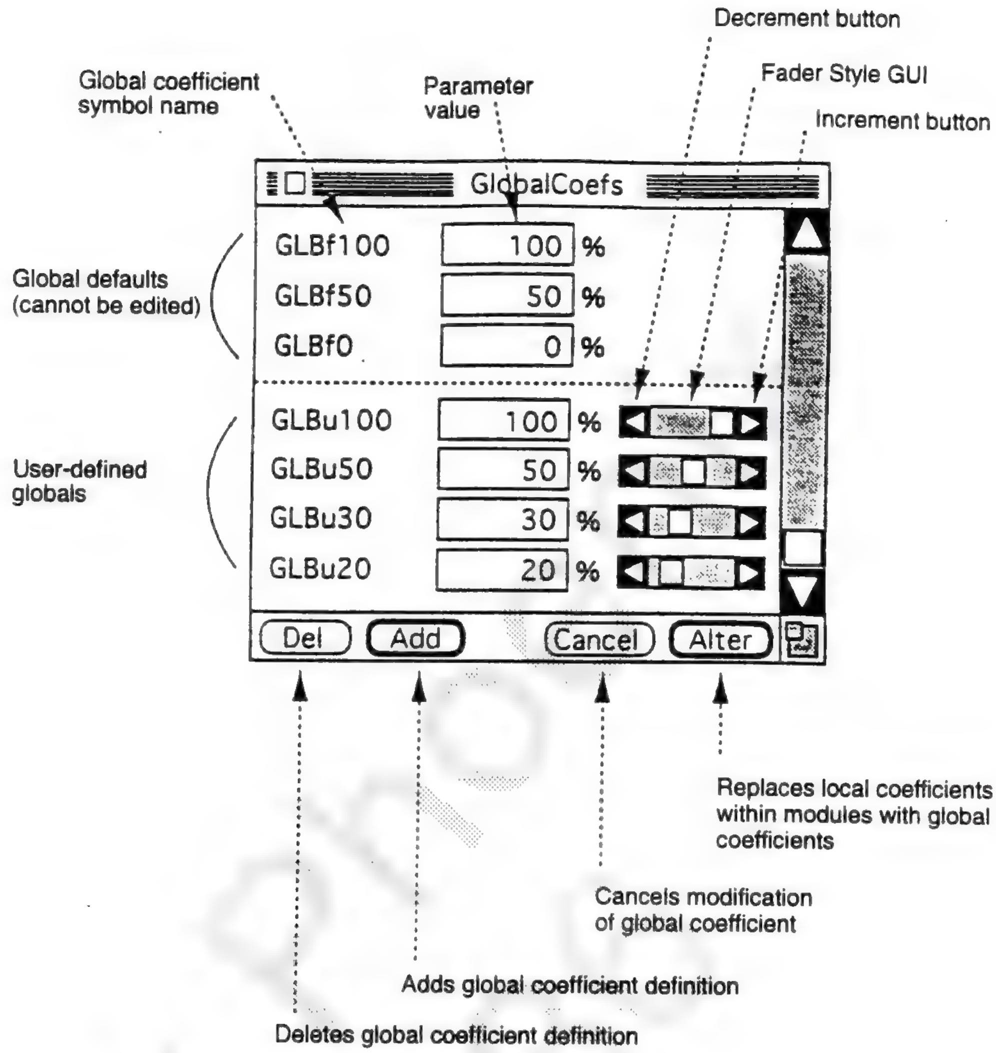


Figure 4 **Global Coefficient screen**

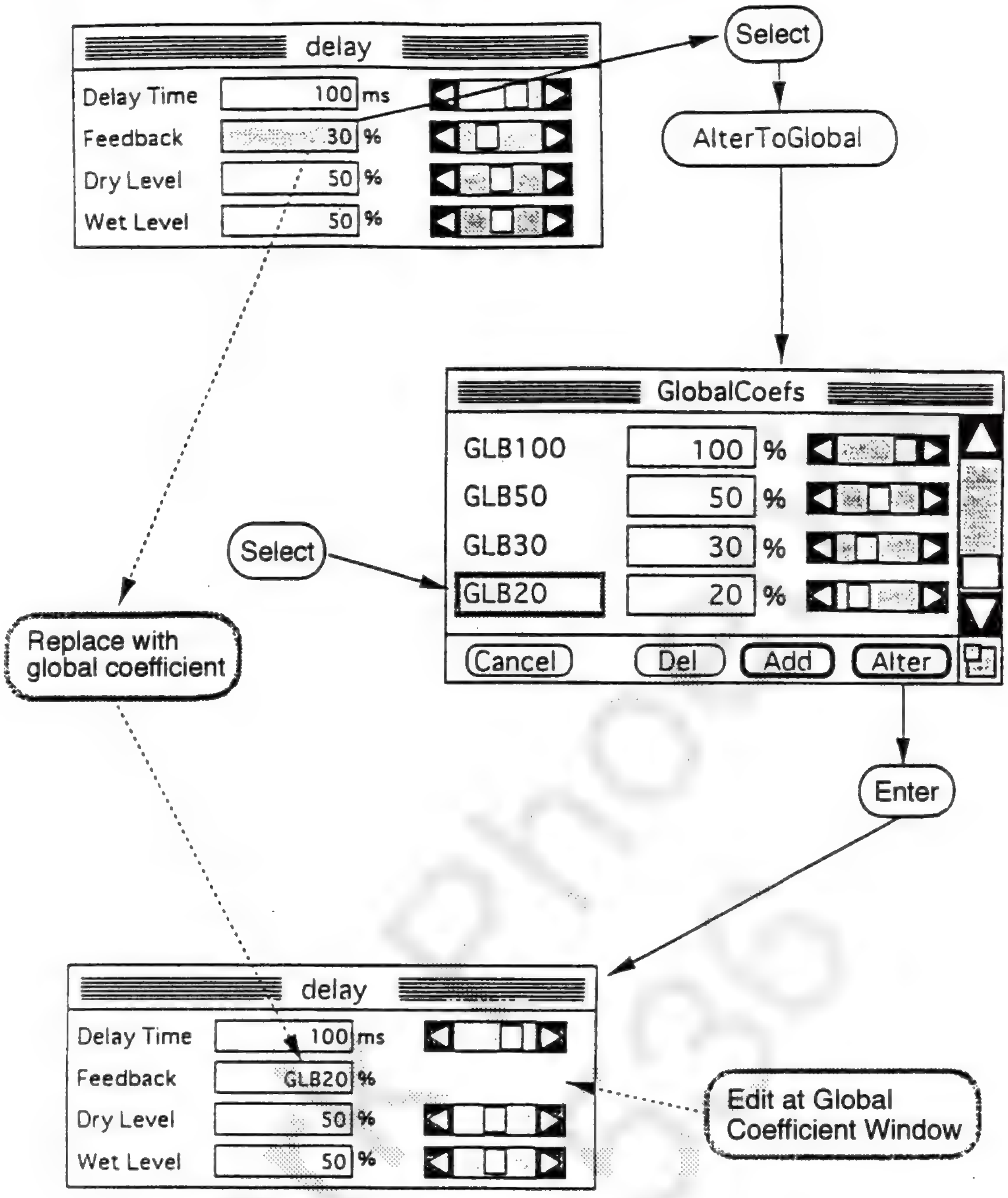


Figure 5 **Using Global Coefficients**

8.0 Addendum to Linker Specifications v1.10

#0 Exclusive Areas in the Target's Memory

#1 Adding a Header to Execute Format Files

#2 Download Method

- ## Write the file (program) name in the header to **TrgtMem_Filename**, which is one of the dedicated areas in the target's memory. (This is applicable only when downloading the file to the main DSP unit.)
- ## Loading multiple DSP programs.

#3 Displaying Decimal Numbers in the Module GUI Screen

#4 New Dialog Box for the RBLen Setup in the Linker Algorithm Edit Screen

#5 Displaying Link Results

#6 Managing Coefficient Tables

##Implementing a Dynamic Filter
##EG/LFO

Notations Used in this Document

HeaderSize: Size in bytes of the header included in the execute format file. = 40H [bytes]

MicroProgSize: Size of the microprogram. 100C00H - 100700H = 500H [bytes] (The size includes the gap between MADRS and MPRO.)

NumberOfCoefTables: Number of coefficient tables that are linked in the execute format file.

CoefTableSize: Size of one coefficient table. 100H [words] x 2 x 5 = A00H [bytes]

RBLen: This is a parameter that is set with the Linker to define the ring buffer size. The allowed values are 0 to 3. When a file is downloaded, this value is set in **RBL [1:0]**, which is a DSP internal register.

#0 Exclusive Areas in Target Memory

Although various exclusive system areas at fixed addresses in the target's memory will be reserved, the actual addresses have yet to be defined. In lieu of actual address numbers, the following labels have been defined to represent the starting address of each memory area in this document.

TrgtMem_DSPprogAddress
This target memory address marks the beginning of a multi-byte area into which the Linker sets the first address of the DSP program (execute format file) download area.

TrgtMem_DSPprogSize
This target memory address marks the beginning of a multi-byte area into which the Linker sets the size of the DSP program (execute format file) download area. (This address is valid only for the spare program area.)

TrgtMem_DSPregisters
This target memory address marks the beginning of the address to which the DSP registers have been mapped. This address is already set by the hardware, and its value is 100700H.

TrgtMem_Filename
This target memory address marks the beginning of a 20H-byte area into which the file (program) name that is currently loaded in the main DSP unit is written.

TrgtMem_RBP
This target memory address marks the beginning of a one-word area in which the Linker sets the upper seven bits (i.e., **RBP [19:13]**) of the first address in the area used by the DSP as external RAM. This address is already set by the hardware, and its value is 100402H. (The format in this one-word area is described in the SCSP hardware specifications.)

TrgtMem_RBL
This target memory address marks the beginning of a one-word area in which the Linker sets the size bit **RBL [1:0]** of the ring buffer area. The ring buffer is in the area used by the DSP as external RAM. This address is already set by the hardware, and its value is 100402H. (The format in this one-word area is described in the SCSP hardware specifications.)

TrgtMem_DSPRAMSize
This target memory address marks the beginning of a multi-byte area in which the Linker sets the external RAM size that is currently enabled for use by the DSP. The DSP can use this RAM area, which is defined by the RBP position stored in Trgt**Mem_RBP** and the size specified by **TrgtMem_DSPRAMSize**, in the following ways: a ring buffer area, a reserved area (20H words), or a coefficient table area.

Note:

In this addendum, two notations are used to present label names. One notation refers to a previous label and encloses the label name in parentheses, as in the following example:

```
(TrgtMem_...)
```

This notation denotes the contents of the memory address specified by the label.

The other notation uses the label name by itself, as follows:

```
TrgtMem_...
```

This notation indicates the value of the memory address specified by the label.

#1 Extension Format Used Exclusively by the Linker in the Execute Format Files

The Linker extends the format of execute format files as shown below. This format can only be used by the Linker, and the extension ".EXL" is added to the file name.

Organization of the Extension Format

Header
Microprogram
Coefficient table #0
Coefficient table #1
Coefficient table #2
. . .
#(NumberOfCoefTables - 1)

Header Contents

File (program) name → character string (up to 31 characters)
Requested ring buffer size: **RBLen**
Number of coefficient tables (**NumberOfCoefTables**)
Spare area

Microprogram

(Text format)

Coefficient Table Data

Coefficient table data consisting of 500H words per table is linked in the extension format. This data is used to implement dynamic filtering.

One word (two bytes) of data is represented in hexadecimal notation (four digits) as xxxx. Each line contains eight words that are separated by spaces. This pattern is repeated for 160 lines. These 160 lines form the 1280-word (500H-word) data block for one coefficient table. This data block is repeated for each table with a blank line separating each of the blocks. An example of this data is shown in the Execute Format Text File Extension Format item on the next page.

Execute Format Text File Extension Format	
File Contents	**Comment**
	<u>Header</u>
FILENAME="Sample.EXL"	← File name
RBLEN=x	← RBLen value. x = 0 to 3
NUMBEROFCOEFTABLES=x	← Number of coefficient tables
COEF	<u>Body of microprogram</u>
00:xxxx:*coefname00*	
01:xxxx:*coefname01*	
↓	
ADRS	
00:xxxx:*addressname00*	
01:xxxx:*addressname01*	
↓	
PROG	
00:xxxx xxxx xxxx xxxx	
01:xxxx xxxx xxxx xxxx	
↓	
7F:xxxx xxxx xxxx xxxx	
COEFTABLE	<u>Coefficient Table</u>
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #0 word000 to word007
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #0 word008 to word00F
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #0 word010 to word017
↓	↓
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #0 word4F8 to word4FF
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #1 word000 to word007
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #1 word008 to word00F
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #1 word010 to word017
↓	↓
xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx	Table #1 word4F8 to word4FF
↓	**Note:** Repeat up to table #(NumberOfCoefTables - 1) below

Binary Extension Format

The Binary Extension Format is used when the contents of execute format text files in extension format are converted into binary and downloaded into the target's DSP spare program area (explained later). The target's CPU can then call up the files individually and transfer them into the DSP.

When an execute format text file in extension format is downloaded, it will exist only in the target's memory after the Linker converts the file to the Binary Extension Format.

Binary Extension Format

FILENAME	20H [bytes]
RBL	1 [byte]
NUMBEROFCOEFTABLES	1 [byte]
(reserved)	1EH [bytes]
DSP Micro Program	MicroProgSize = 500H [bytes]
Table #0	CoefTableSize = A00H [bytes]
Table #1	CoefTableSize = A00H [bytes]
↓	
Table #(NumberOfCoefTables - 1)	CoefTableSize = A00H [bytes]

#2 Downloading an Execute Format File (Extension Format)

The memory management system in the target presets the (**TrgtMem_DSPprogAddress**) value. The Linker refers to this value when it downloads the execute format file, then it executes one of the following operations:

(A) Download the file to the main DSP unit.
(B) Download the file to the DSP spare program area (in the target's memory).

The criteria for selecting which operation to execute is as follows:

If (TrgtMem_DSPprogAddress) = TrgtMem_DSPregisters	(A)
If (TrgtMem_DSPprogAddress)! = TrgtMem_DSPregisters	(B)

Operations for each case are explained below.

Downloading the File to the Main DSP Unit

0) The Linker checks to see whether the size of the area to be used by the DSP as external RAM is adequate.

(TrgtMem_DSPRAMSize) ≥ 2^RBLen x 4000H + 20H x 2 + NumberOfCoefTables x CoefTableSize

If this condition is not satisfied, the Linker outputs an error message and stops the download.

Notes:

1. For **RBLen**, the Linker refers to **RBLEN** in the extension format header.
2. For the **NumberOfCoefTables** value, the Linker refers to **NUMBEROFCOEFTABLES** in the extension format header.

1) The Linker writes the file (program) name stored in the header to the exclusive area **TrgtMem_Filename** in the target memory.

2) The Linker writes the **RBLen** stored in the header to **RBL [1:0]**, which is located in exclusive area **TrgtMem_RBL** of the target memory.

3) The Linker fetches the microprogram from the format of the execute format file and downloads that portion to the area that starts from the **TrgtMem_DSP** registers (main unit of the DSP hardware).

4) The Linker fetches the coefficient table portion from the format of the execute format file and downloads it to the following address:

```
First address of the coefficient table area = RBP [19:13] x 2000H +
2^RBLen x 4000H + 20H x 2
```

If several coefficient tables are linked, all of the table areas are downloaded as a set.

Notes:

1. **RBP[19:13]** is found in the exclusive area **TrgtMem_RBP** of the target memory. RBP [19:13] is multiplied by 2000H to fill the omitted bits of RBP [19:13] (bit 12 to bit 0) with zeros and convert RBP [19:13] to the real address.

2. "2^RBLen" represents two raised to the RBLen power. "2^RBLen x 4000H" is the ring buffer size.

3. The addition of "20H x 2" reserves a 20H space area after the ring buffer. The area will be used to transfer Software EG/LFO data (addition scheduled after January 1994).

4. The example given below shows the calculation of the first address of the coefficient table area.

   RBP [19:13] = {0, 1, 0, 0, 0, 0, 0}
   RBLen = {1, 0}

   When the above settings are used, the first address of the coefficient table area is calculated as follows:

   20H x 2000H + 2^2H x 4000H + 20H x 2 = <u>50040H</u>

Downloading the File to the DSP Spare Program Area

0) The Linker uses the following conditional argument and checks whether the download destination has sufficient space available. If this condition is not satisfied, the Linker outputs an error message and stops the download.

(TrgtMem_DSPprogSize) ≥ HeaderSize + MicroProgSize + NumberOfCoefTables x CoefTableSize

Note:

For the NumberOfCoefTables value, the Linker refers to NUMBEROFCOEFTABLES in the extension format header.

1) The Linker refers to (TrgtMem_DSPprogAddress) and calculates the download destination address.

2) The Linker downloads the execute format file in binary format (see #1).

#3 Displaying Decimal Numbers in the Module GUI Screen

The GUI of the input/output modules (I/O modules) shows the number of the currently connected **MIXS** and **EXTS** (for the input module's "Input") and the number of the currently connected **EFREG** (for the output module's "Output"). (See Figure 6, "Module Object GUI screen"). The numbers are displayed in decimal notation. The number display allows the checking of values without opening the Module Parameter Edit screen.

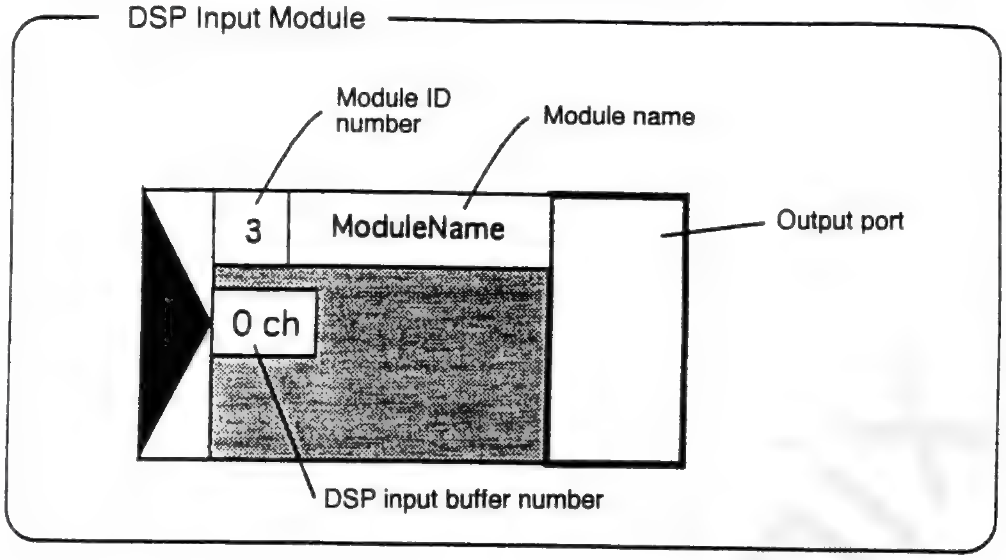
The displayed values are as follows:

1. Input

   The displayed values are from 0ch to 17ch. These values conform to the range described for the Source parameter mentioned in the *SCSP/DSP Effect and I/O Module Specifications* user parameter list.

2. Output

   The displayed values are 0ch to 15ch. These values conform to the range described for the **Destination** parameter mentioned in the *SCSP/DSP Effect and I/O Module Specifications* user parameter list.



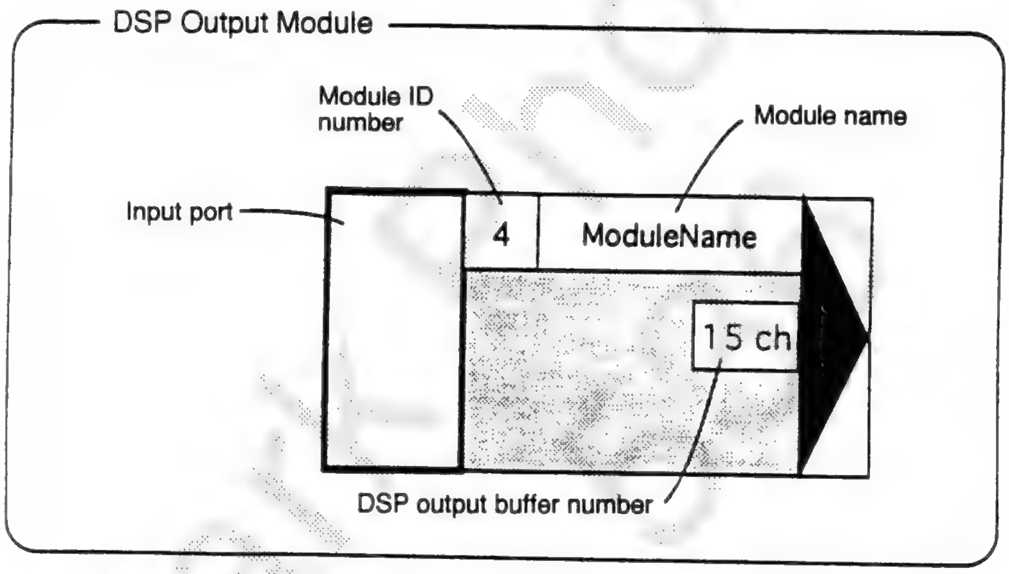


Figure 6 Module Object GUI screen (1)

#4 New Dialog for the RBLen Setup in the Linker Algorithm Edit Screen

Set the size of the ring buffer (RBLen) to be used by the algorithm being created. For this purpose, a new Setup Menu will be added and it will include RingBuffer as a menu item. Selecting this menu item opens the Ring Buffer Size Setup dialog box. The Linker will have RBLen = 0 as the default value. If there is no value in this dialog box, the default value will be used.

The RBLen value that is set in the dialog box is used by the Linker for the following purposes:

1) When allocating the delay RAM during Link command execution, the Linker displays an error message and stops the link operation if the total delay RAM used by the modules exceeds the ring buffer size that is calculated from the RBLen setting.

2) In modules that use coefficient tables, the Linker must deal with the address values of the coefficient table area, which is located after the ring buffer area. The first address of the coefficient table area changes according to the size of the ring buffer. When the first address of the coefficient table area changes, the address of each coefficient table also changes. At Link command execution, the Linker refers to the RBLen setting and determines the address of each coefficient table.

3) At Download command execution, the Linker writes the RBLen value that was specified in the dialog box to the internal DSP register RBL [1:0], which is located in address TrgtMem_RBL of the target memory.

Notes: RBL [1:0]

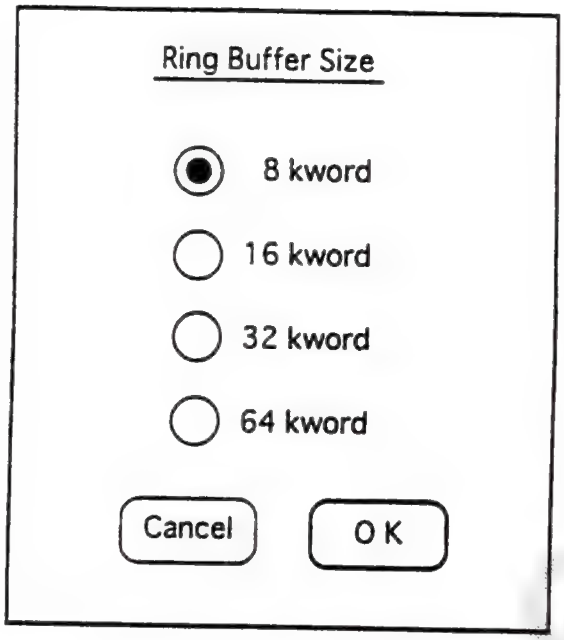
RBL [1:0] is located in address TrgtMem_RBL of the target memory. The DSP hardware refers to this register to find out the ring buffer size, then it allocates the ring buffer area accordingly.

If RBL [1:0] is set to {1, 1}, the format cannot have a coefficient table area.

The correspondence between the RBL [1:0] setting and the ring buffer size is as follows:

RBL[1:0]	Ring buffer length
0	8 [kwords]
1	16 [kwords]
2	32 [kwords]
3	64 [kwords]

Ring Buffer Size Setup Dialog



#5 Displaying Link Results

After using the Link command (Process Menu - Link) to execute the link operation, the Linker opens a dialog box that shows information on the number of hardware resources used in the current effect structure. The dialog box includes the following items:

1) Number of program steps
2) Amount of coefficient RAM used (number of words)
3) Amount of address constant RAM used (number of words)
4) Amount of DRAM used for the ring buffer (total delay RAM size for all modules)
5) Amount of DRAM used for coefficient tables (20H word reserved area + coefficient table length x number of coefficient tables)
6) Size of DRAM area that must be allocated for the DSP, that is, the beginning of the area described in 4) to the end of the area described in 5). This value differs from the total of 4) and 5), and it is determined as follows:
   [ring buffer area calculated from the RBLen setting] + [area for "5)"]

#6 Managing Coefficient Tables

The addition of a Coefficient Table Allocator will enable the Linker to handle coefficient table data. The Coefficient Table Allocator may be set up as a modeless window that can always be left open for reference. This window is opened by selecting the CoefTable sub-menu item in the Window menu. (CoefTable will be set up as a new item under the Parameters menu item.)

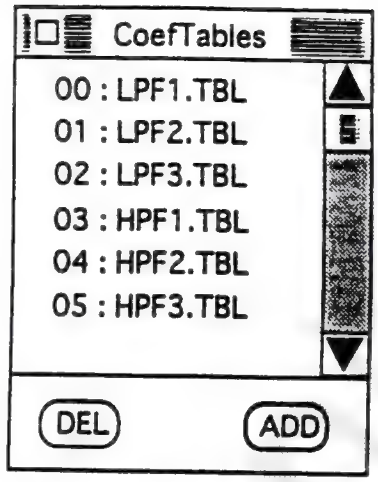
The Coefficient Table Allocator manages the coefficient table files that are used by the algorithm being edited. The filenames of the currently registered coefficient table files are displayed in the window with numbers that indicate the order in which the files were registered (see “Coefficient Table Allocator screen” on page 33).

ADD button
Opens a standard file dialog box and registers a new coefficient table file. The new file is added after the last coefficient table file that was registered.

DEL button
Deletes the registration of the specified coefficient table file.

Coefficient Table Files
A coefficient table file stores binary data having a size of A00H. The file extension is ".TBL".

Coefficient Table Allocator Screen



9.0 Implementing a Dynamic Filter

At the current time, the methods for implementing a dynamic filter are as follows:

1) FEG Generation Method
   Use the sound source slot to produce FEGs. Use the Sound Editor to edit FEGs.

2) Coefficient Table Allocation
   From the coefficient table allocation screen, allocate a coefficient table by selecting the coefficient table file.

3) Editing the Dynamic Filter Module

   a) Setting up a FEG
      To set up a FEGs for the Dynamic Filter module in the mixer screen by setting the sound generator's FEG slot assignment destination (set in the Mixer screen) to the same channel as the Dynamic Filter module's DSP Modulator Input Buffer.

   b) Selecting a Table
      To select a table, enter the table number.

SEGA OF AMERICA, INC.
Consumer Products Division

Sound Development Manual Outline

Doc. # ST-81-012494

December 5, 1993, Ver. 1.00

1.0 System Outline

Hardware Requirements

Device	Model	Description
Development host machine	Macintosh	Macintosh II series or later with SCSI interface. Operating system: KanjiTalk 7 or System 7 or later RAM: 16 MB or more HDD: 300 MB or larger and 1 GB when making HD recordings
Saturn Target box		Sound board is capable of operating alone.
MIDI instrument	MIDI keyboard, etc.	Instrument with MIDI output capability Used for tone development, composition, and creating sound effects
Audio equipment	CD player, DAT, etc.	Device capable of digital output Used for editing waveforms and HD recording
MIDI interface	Studio 5, etc.	MIDI interface for Macintosh

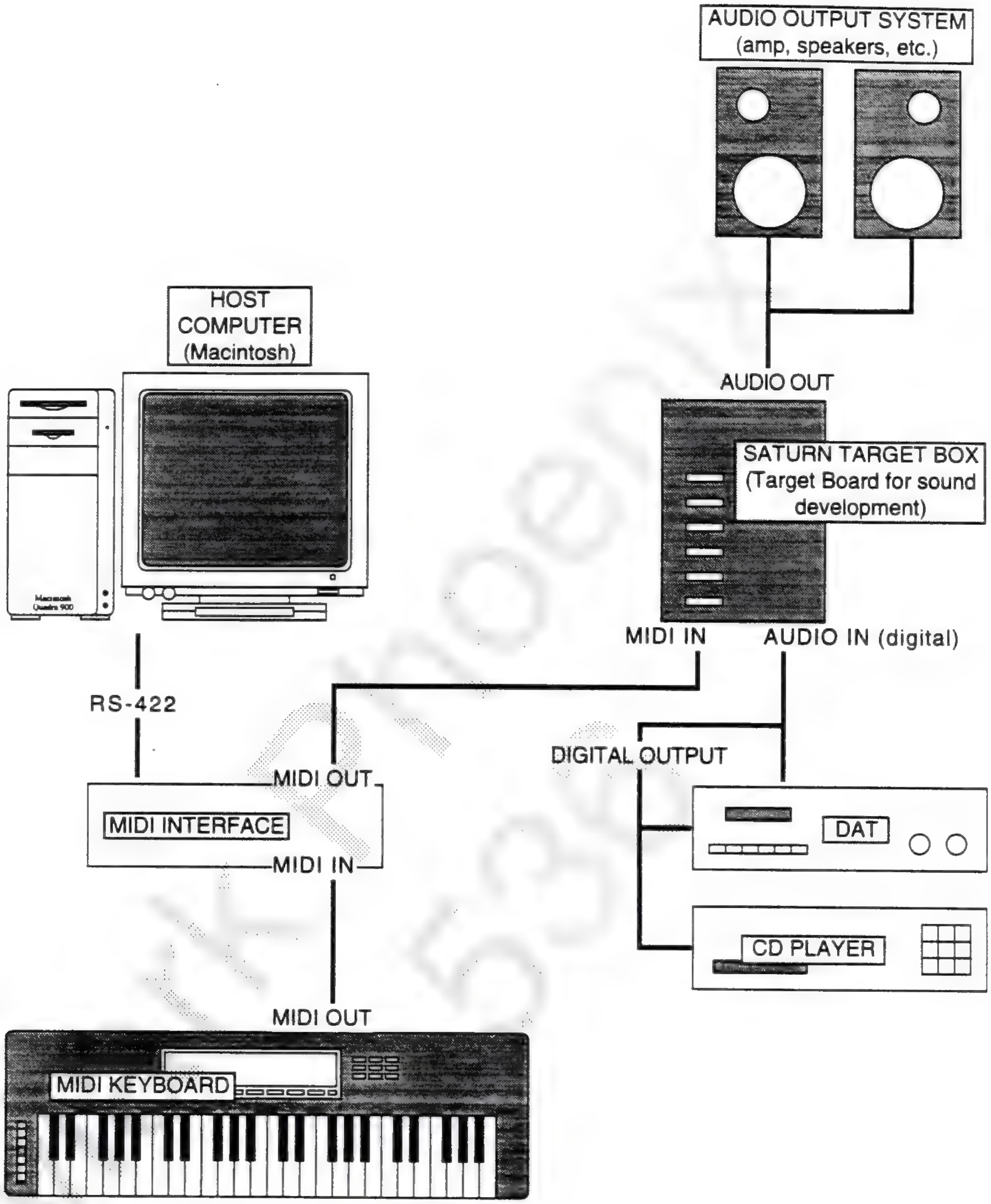
Software Requirements

System	Software	Source	Description
Tone development tool	Waveform editor Tone editor DSP linker	SEGA Market item SEGA SEGA	Used to edit waveforms and HD recordings Waveform edit tools available on the market can also be used (devices that support AIFF format) FM/PCM tone development DSP program development
Composition tools	MIDI sequencer	Market item	Digital Performer (Mark of the Unicorn) Studio Vision (Opcode Systems) Cubase Audio (Steinberg), etc.
Sound development support system	Master system	SEGA	Map Tool Hardware Simulator Optimization Linker

The tools listed below are for writing and changing sound drivers and other programs, and are not required for the development of sound data (tunes and sound effects).

System	Software	Source	Description
Program development tools	Text editor Assembler Linker Debugger	SEGA Market item SEGA SEGA SEGA	Use for preparing programs, data, etc. Word processors and text editors available on the market can also be used (those that support TEXT format) Macro assembler for 68000 Linker for 68000 Remote debugger for 68000

Hardware Setup for Development



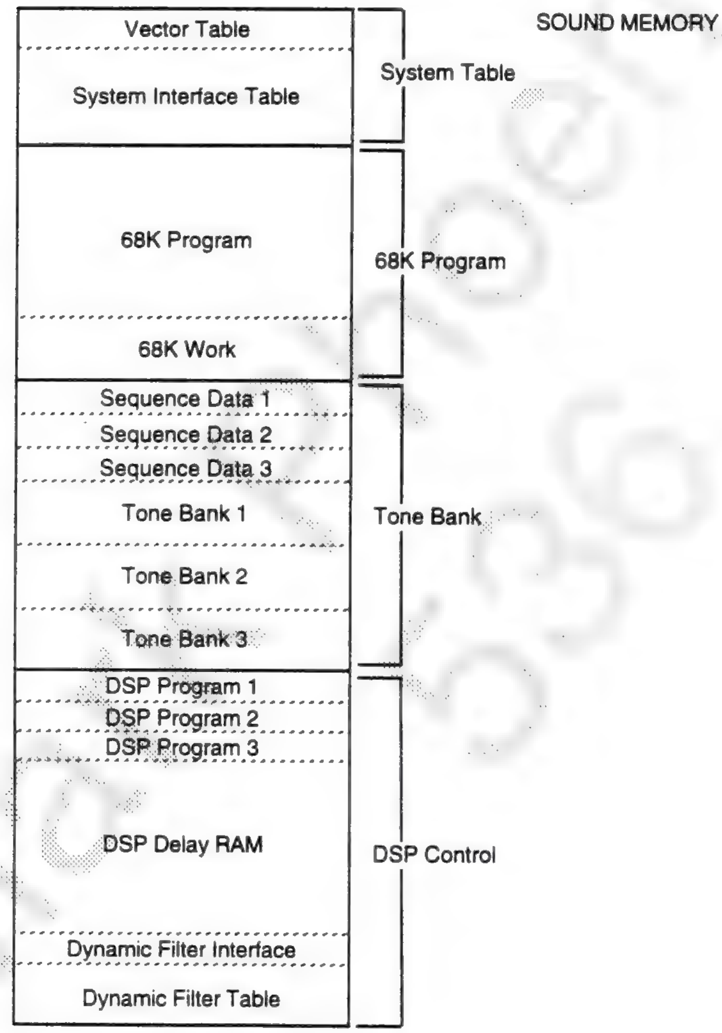
2.0 Development Overview

The Saturn sound system makes it possible to produce sounds without knowledge of assembly language or other computer programming by comprising the Saturn itself as a MIDI-generated 32-voice polyphonic multiple sound source. Therefore, game sounds can be produced by preparing "tone bank data," which become the Saturn sound source, and then composing in a desktop music or computer music environment using a MIDI sequencer available on the market. Also, by using a DAT or CD player with a digital output for obtaining PCM waveforms, which required a sampler or other device in the past, these sounds can be obtained from the Saturn Target Board. Development is made more efficient by allowing editing on a Macintosh.

Sound Data and Sound Memory

Sound data are categorized in the following four types, and multiple data can be held outside the sound area map as required by the game scene. The memory arrangement of each of these scenes of multiple data is determined by the sound area map. Each of the sound data can be freely replaced according to the setting of the sound area map.

- Sound area map: Map data in which the arrangment of the tone bank data in the sound memory, the DSP program, and the sequence data is set according to each scene of the game.
- Tone bank data: One independent tone block in which waveform data, layer data, and multiple voice data are stored.
- DSP program: DSP execution file linked to one required effect module.
- Sequence data: Tune and sound effect data created from the data of one tone bank.



Development Procedure

Startup of Master System
*Memory mapping and data transfer

Creating tone bank data
*Obtaining waveforms
*Editing waveforms
*Editing tones
*Creating tone banks
*Using tone libraries to be supplied by SEGA
*Transferring to the Saturn board

Creating DSP programs
*Selecting desired effects from the DSP library
*Adjusting parameters as required
*Creating new effects as required
*Transferring to the Saturn board

Composition and arrangement
(Creating sequence data)
*Composing and editing by means of a sequencer
*Converting to a standard MIDI file
*Transferring to the Saturn board

Creating sound effect tones
*Creating sound effect tones like creating tones
*Creating sound effect tone banks
*Transferring to the Saturn board

Creating sound effect sequences
*Creating sound effects by controlling sound effect tones with a sequencer
*Converting to a standard MIDI file
*Transferring to the Saturn board

Assembling games
*Simulation (to be implemented in the next version)
*Hardware simulator
*Optimizing linker
(Currently not supported, but will be supported in future versions.)

Install in actual game

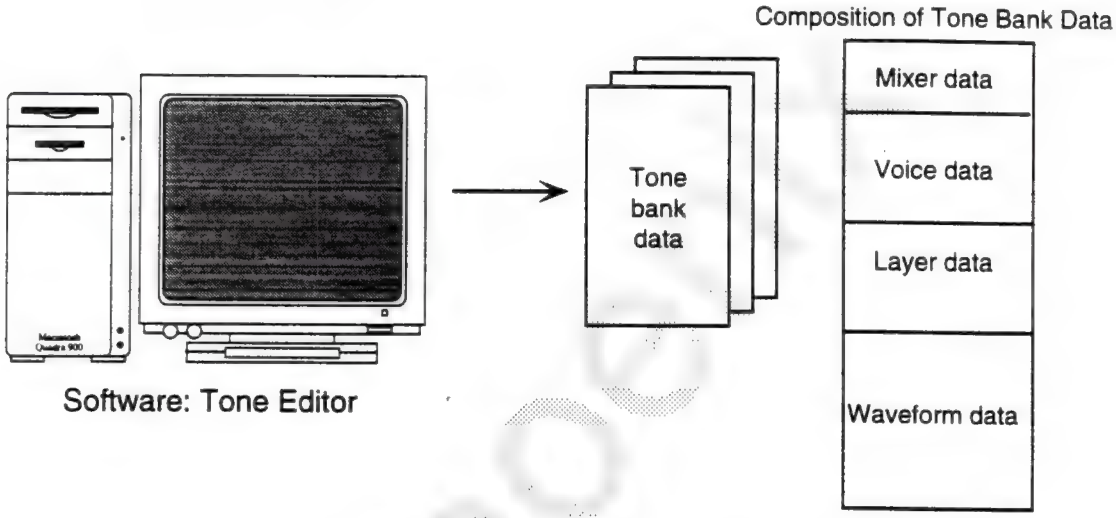
- This procedure is repeated for each scene of the game.
- Sound data are made by repeating these operations.
- The order of the steps may be changed except for the allocation of map tools

Starting the Master System

When developing sound, first start up the Master System, then transfer the sound area map of the sound to be created and the necessary data to the Target board. It is more convenient to create the sound area map in advance using the Map Tool, but it can also be added or modified as necessary.

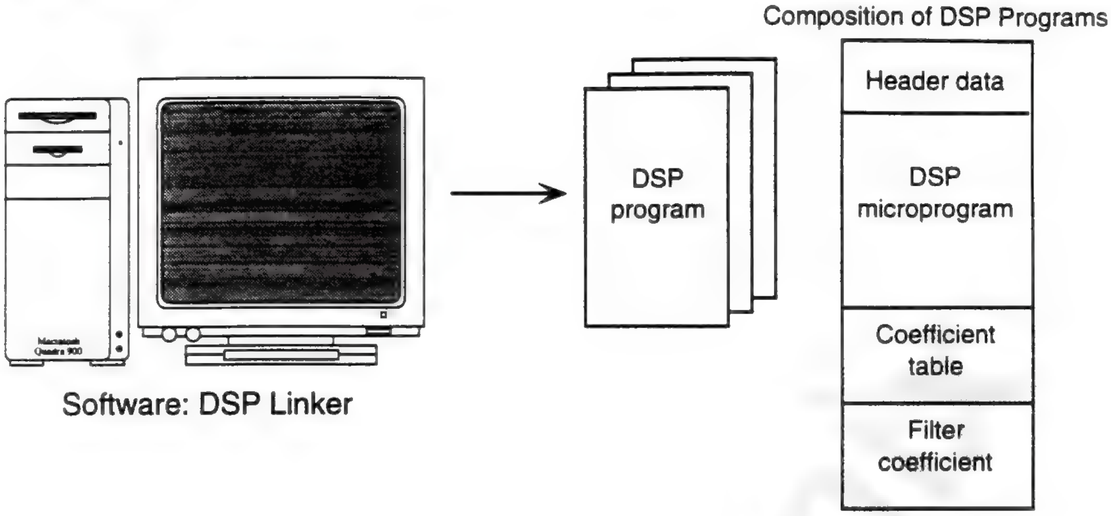
Creating Tone Data

When creating sound data, the first thing required is tone data. Data blocks (referred to as tone bank data) can be made using the tone editor.



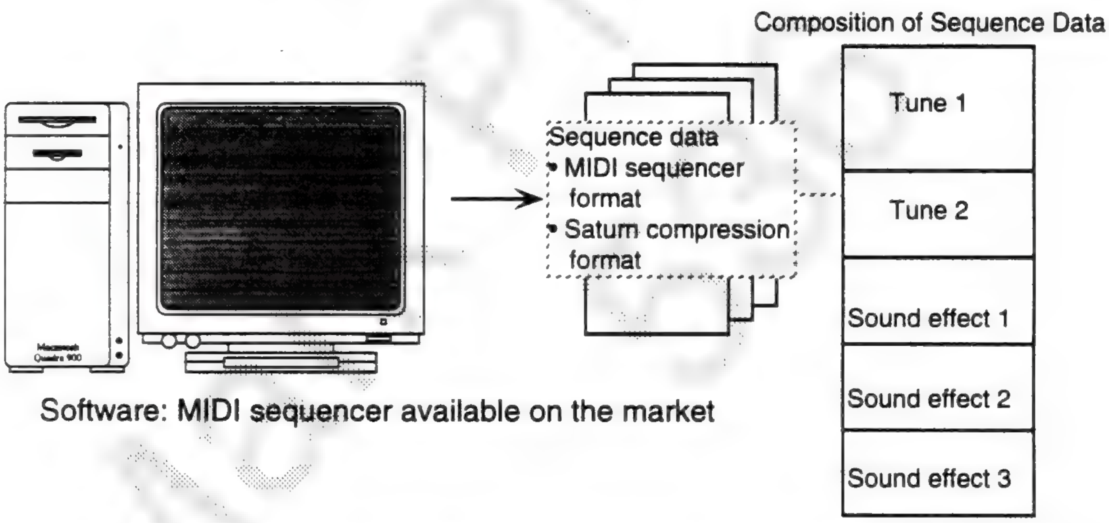
Tone bank data make up an independent tone by combining mixer, voice, layer and waveform data. One tone that uses a PCM waveform or FM is a "layer", and one voice can be produced from one layer or by combining multiple layers. Voices correspond to program changes in MIDI, and one bank can hold up to 128 voices. Therefore, up to 128 musical instruments can be played with one bank of tone data.

Creating DSP Data
When a sound effect using DSP is required, a DSP program is created by utilizing the DSP linker.



Reverb, echo, chorus and other modules are provided in libraries. Select the items required from among those modules and link them. Multiple effects can be used simultaneously on the Saturn sound system, but do not exceed 128 steps overall. For example, an echo iof 20 steps, a chorus of 22 steps and an equalizer of 5 steps, add up to 47 steps.

Creating Sequence Data
If tone bank data has been created, sequence data can be made. Sequence data is a data file comprising tune and sound effect data, and is made up of MIDI note data.



There are two types of sequence data: tune data created on a MIDI sequencer, and Saturn format sequence data in which the tune data is compressed and can be loaded into sound memory. The Saturn format data and MIDI data are essentially the same, with the only difference being whether the tune data is compressed or not.

Sound Source Specification for the Saturn Sound System

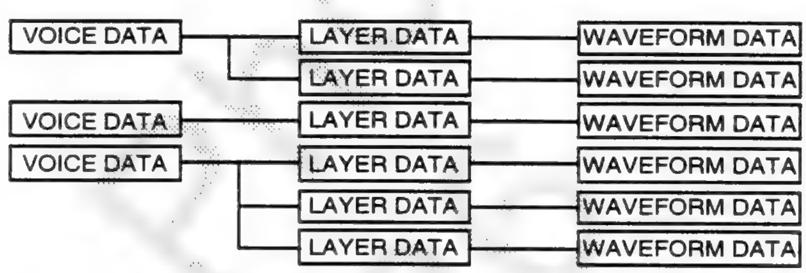
The Saturn sound source is a 32-voice polyphonic multiple sound source corresponding to the 32 channels (32 tracks) of MIDI. This means:

- 32-voice polyphonic multiple sound source: can generate a maximum of 32 sounds simultaneously as a system.
- 32 channels of MIDI: a multiple sound source capable of playing up to 32 instruments at once.

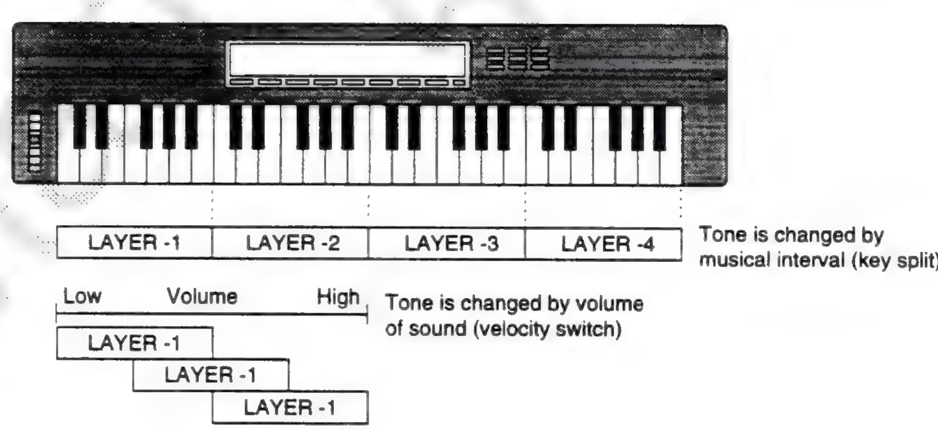
The data for each instrument (tones) is created as tone bank data, which comprise waveform data, layer data and voice data. The way these data are comprised makes it possible to achieve variations of various instruments (tones).

Voices and Layers

The Saturn sound source has 32 sound slots in its hardware, whereby it is able to generate 32 sounds simultaneously. The tone setting for one of these sound slots to called a layer. This is the most basic data module of the sound source section, and various tones are produced using all these layers as a basis. One tone produced by using one or several of these layers comprises voice data, which is equivalent to one instrument sound corresponding to a MIDI program change. In other words, one instrument sound can be produced by using several different tones. Voice data comprise the settings that indicate which and how many layers are used, and which layers are played according to changes in the musical interval and volume.

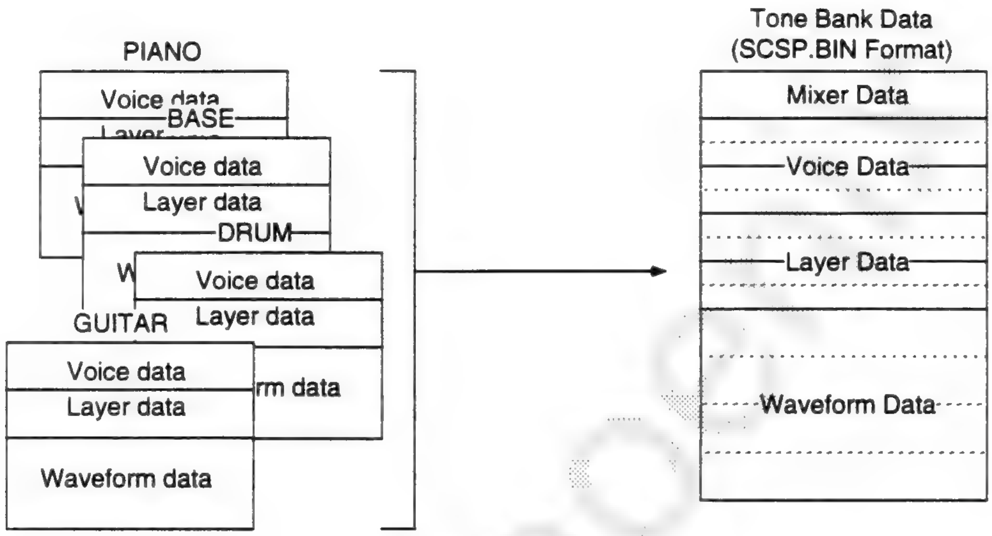


Using this function, various instruments and tones can be created. Depending on the musical interval, they can be separated into several tones (key split), and the tone can be easily changed depending on the volume of the sound (velocity switch).



Tone Bank Data

A bank of tone data make up a data block in which voice data for the required instruments (tones) are compiled. One bank of tone data have at least one voice and can have up to 128 voices if memory allows. In the Saturn sound system, there can be multiple tone banks in the same map, and each can be generated simultaneously as independent sound sources. Therefore, it is possible to insert another bank while playing tunes or sound effects in one bank. This makes possible a flexible system with good development efficiency and memory utilization.



MIDI Channels and Voices

The Saturn sound system can handle up to 32 channels (tracks) of MIDI data at once, and can handle up to 32 instruments simultaneously, assuming one MIDI channel per instrument. However, all tunes and sound effects of these must be played. This is not a problem in scenes where tunes or sound effects are played individually. In scenes where one or more tunes and sound effects are played simultaneously, each must be allotted to a different channel. In this system, the 32 MIDI channels are assigned and played dynamically.

Channel	Voice Assignment
ch1 Bank#	Voice#
ch2 Bank#	Voice#
ch3 Bank#	Voice#
ch4 Bank#	Voice#
ch5 Bank#	Voice#
ch6 Bank#	Voice#
ch7 Bank#	Voice#
ch8 Bank#	Voice#
ch9 Bank#	Voice#
ch10 Bank#	Voice#
ch11 Bank#	Voice#
ch12 Bank#	Voice#
ch13 Bank#	Voice#
ch14 Bank#	Voice#
ch15 Bank#	Voice#
ch16 Bank#	Voice#
ch17 Bank#	Voice#
ch18 Bank#	Voice#
ch19 Bank#	Voice#
ch20 Bank#	Voice#
ch21 Bank#	Voice#
ch22 Bank#	Voice#
ch23 Bank#	Voice#
ch24 Bank#	Voice#
ch25 Bank#	Voice#
ch26 Bank#	Voice#
ch27 Bank#	Voice#
ch28 Bank#	Voice#
ch29 Bank#	Voice#
ch30 Bank#	Voice#
ch31 Bank#	Voice#
ch32 Bank#	Voice#

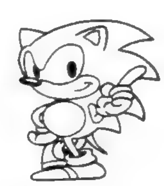
Therefore, each set of sequence data must have this information at the top of the track. In order to play the correct tone, the bank (bank change) and voice (program change) must be specified in each track of sequence data.

APPENDIX

Master System

(12/7/93, Ver. 1.01)

(Currently, only the **Map Tool** is supported. The **Hardware Simulator** and **Optimization Linker** will be supported in the next version.)

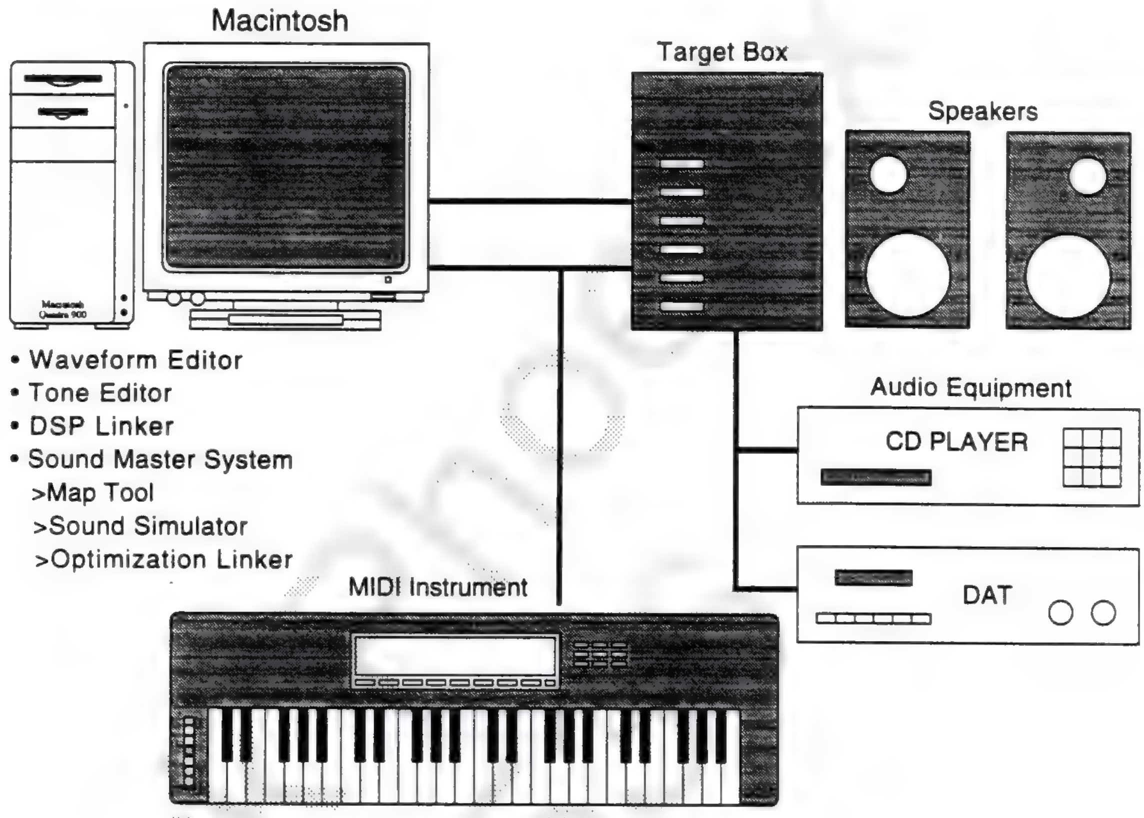


1.0 Map Tool

Outline

The Map Tool creates a "sound area map" used by the sound development system in the Macintosh, and itransfers each type of data according to the sound area map to the Target board when the sound development is in progress. The sound area map shows how the sound memory is used and determines how all of the memory information is used by the waveform edit system, the tone editor, the DSP linker and in other tone development. Therefore, this system must be started first to perform tone development.

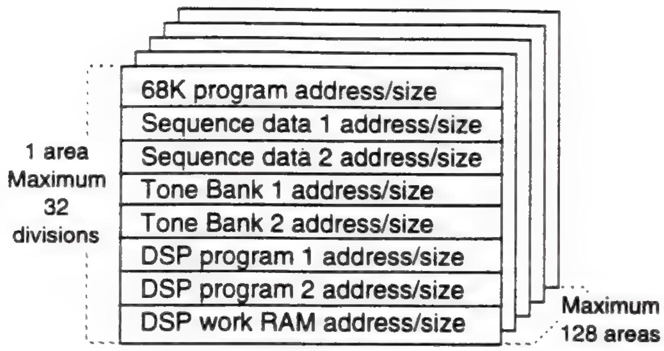
Sound Development System



Sound Area Map and Sound Memory

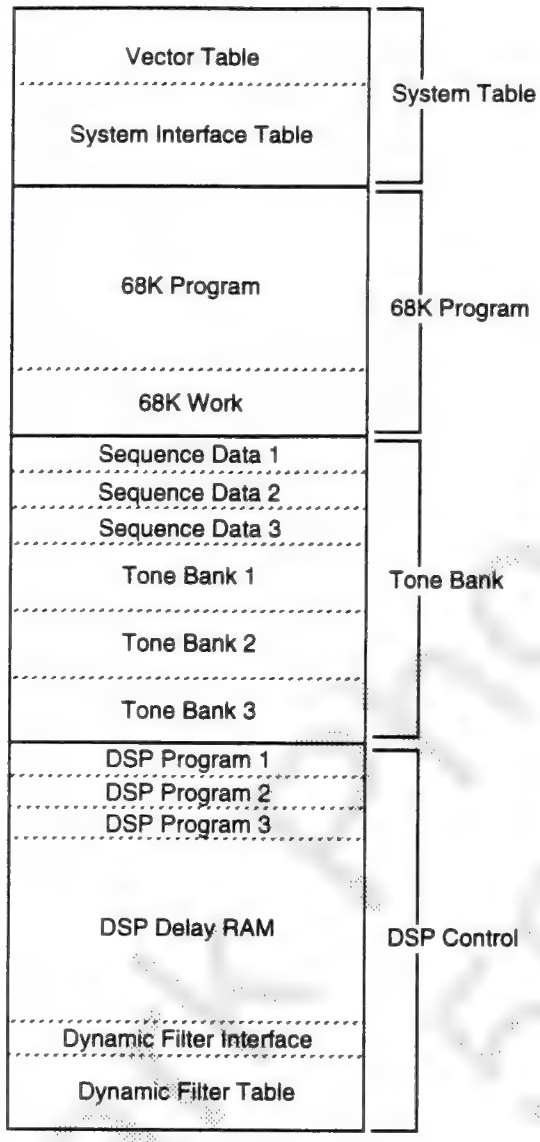
The sound area map is a data file that defines the same number of sound maps for each area (scene or round) of the game as there are areas. The sound map comprises the start address and size of the DSP control area, tone bank area and program area, in the sound memory. This information is referenced by all tone development tools in addition to the sound driver. Therefore, the entire sound development system operates based on this information. The amount of memory in the development system is twice as that in the hardware, but development can be performed by setting the final memory allocation when developing sound data with the initial intent of game assembly, or by freely setting the optimum mapping that takes advantage of the large amount of memory when developing tones, waveforms, or DSP only.

Sound Area Map



Sound Memory

The sound memory is the memory around which the sound development system, which has an 8Mbit capacity (4MBit in hardware), is centered. Therefore, all data, including the programs that control sound, are allocated to this memory. Of these, sequence data, tone banks, and DSP programs can store multiple data simultaneously.



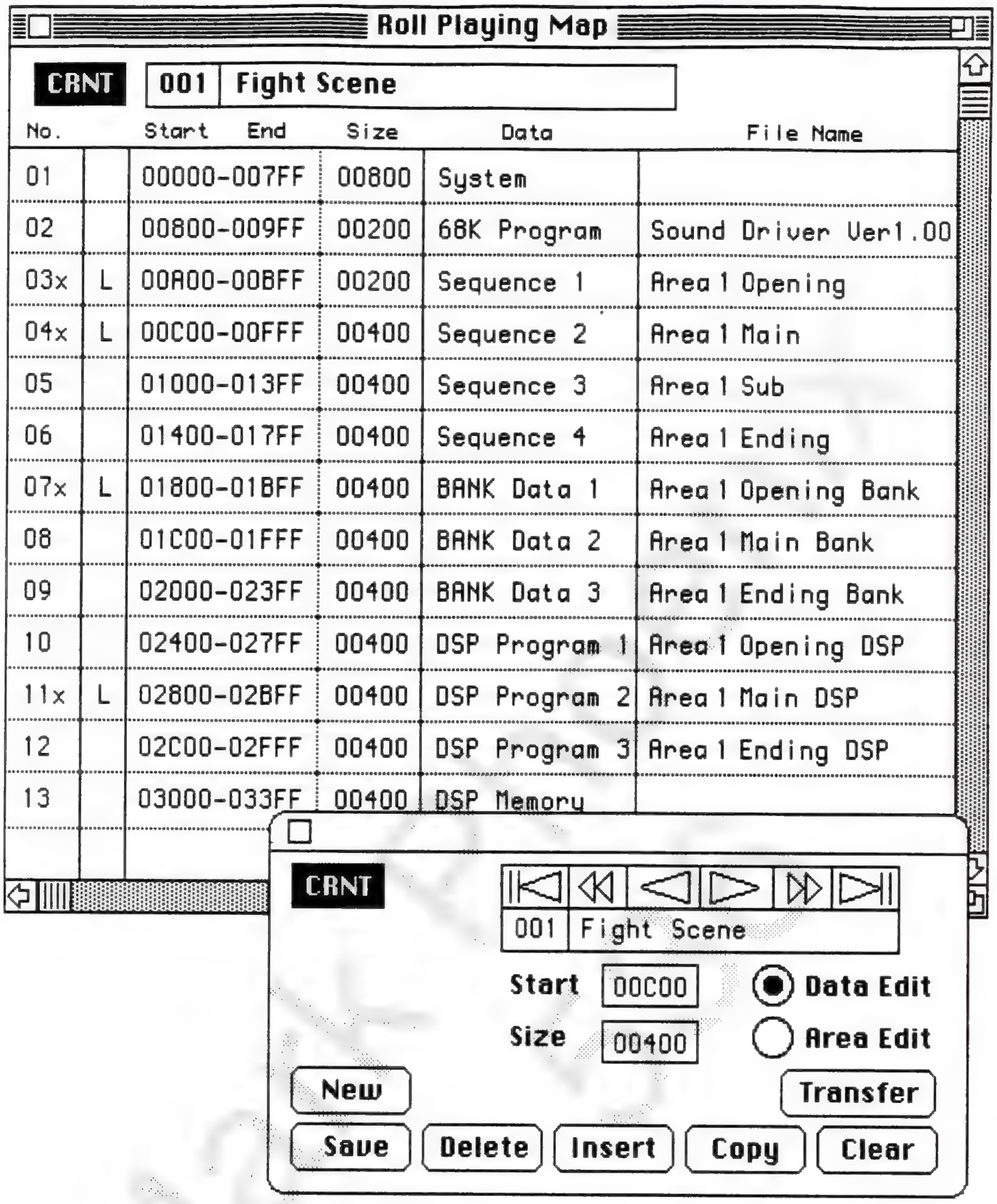
System Table
Since the position of the system table is fixed, it is not recorded in the sound area map. It is used during interface, etc., with the main CPU (SH2).

68K Program
The 68K program comprises a program and data to control the sound CPU and includes the following:
- MIDI/SCSI BIOS
- Main (SH2) interface
- Sound driver
- 68K program work

Tone Bank
The Tone Bank is an area for storing tone bank data and play data corresponding to each area of the game and sound effect data.

DSP Control
DSP control comprises an area where the DSP program is stored, delay RAM which is used and referenced during DSP operation, and a filter table.

Map Edit Screen



X: data has been transferred

L: Indicates automatic transfer at startup

04x	L	00C00-00FFF	00400	Sequence 1	002: Area Opening

CRNT	Appears inverted if the area map being displayed is selected. If it is not selected, click here to select (CRNT) the area being displayed.
|◁ ▷|	Displays the top (bottom) area.
◁◁ ▷▷	Advances (returns) the area four records.
◁ ▷	Advances (returns) the area one record.

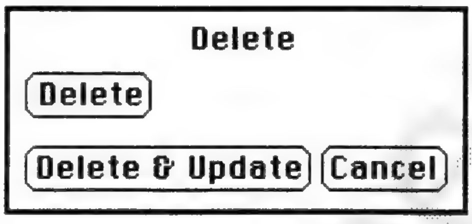
Edit Functions

Two types of edit functions can be selected to edit map data and the entire area. Select one by clicking on **Data Edit** or **Area Edit** in the dialog box. When **Data Edit** is selected, the various map data in the area can be edited. Each of the edit buttons performs its respective processing on the selected map data. Map data can be selected by clicking on the data to edit with the mouse. Multiple map data can be selected at once, with the data displayed becoming inverted when selected.

Data Edit

Delete

When the **Delete** button is clicked, a confirmation message is displayed which allows updating after deletion. The deleted data is stored into memory and can be retrieved by using the **Insert** feature.



Copy

When the **Copy** button is clicked, the selected data is stored into memory and can be retrieved by using the **Insert** feature.

Insert

When the **Insert** button is clicked, the most recently stored data in memory is inserted. The insertion position is in front of the currently selected data, which is displayed inverted, or at the endif no data is selected.

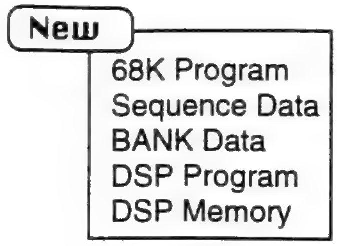
Save

The selected data is also displayed in the dialog box, where it can be edited. Data that can be changed include the start address, size, and three types of read file names. When the **Change** button is clicked, the changed contents are saved.

Clear

When the **Clear** button is clicked, the selected data is deleted (but not the data type).

New
When the New button is clicked, all new data is saved. After the data to be saved is selected from the selection box, the start address and size are input.



Transfer
The currently selected data is transferred to the specified area when the **Transfer** button is clicked. When multiple data is specified, all data is transferred.

Area Edit

When **Area Edit** is selected, the area being displayed can be edited. Each of the edit buttons affects the entire area.

Delete
When the **Delete** button is clicked, a confirmation message is displayed, and the selected area is deleted when the **Yes** button is chosen. The deleted area is stored in memory and can be retrieved by using the **Insert** feature.

Copy
When the **Copy** button is clicked, the selected area is stored in memory and can be retrieved by using the **Insert** feature.

Insert
When the **Insert** button is clicked, the most recently stored data in memory is inserted in front of the selected area. The stored data is not cleared and can be inserted multiple times.

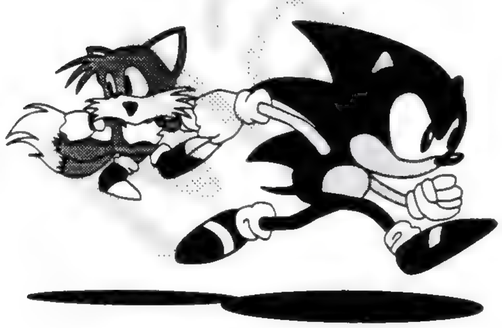
Clear
When the **Clear** button is clicked, the selected area is deleted (but not the data type).

2.0 Hardware Simulator

This section will be supported in the next version.

3.0 Optimization Linker

This section will be supported in the next version.



SEGA™

SEGA OF AMERICA, INC.
Consumer Products Division

Wave Editor
User's Manual

Doc. #ST-99-R1-042594

NOTICE

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.

2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without SEGA's permission.

3. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

4. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

5. No license is granted by implication or otherwise under any patents or other rights of any third party or SEGA Enterprises, Ltd.

6. This document is confidential. By accepting this document you acknowledge that you are bounded by the terms set forth in the non-disclosure and confidentiality agreement signed separately and/in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

(4/20/94 - 001)

READER CORRECTION/COMMENT SHEET

Keep us updated!

If you should come across any incorrect or outdated information while reading through the attached document, or come up with any questions or comments, please let us know so that we can make the required changes in subsequent revisions. Simply fill out all information below and return this form to the Developer Technical Support Manager at the address below. Please make more copies of this form if more space is needed. Thank you.

General Information:

Your Name ______________________________ Phone ______________

Document number ST-99-R1-042594 Date ______________

Document name Wave Editor User's Manual

Corrections:

Chpt.	pg. #	Correction

Questions/comments: ______________________________

____

____

____

Where to send your corrections:

Fax: (415) 802-3963
Attn: Manager,
Developer Technical Support

Mail: SEGA OF AMERICA
Attn: Manager,
Developer Technical Support
275 Shoreline Dr. Ste 500
Redwood City, CA 94065

Wave Editor
User's Manual

1.0 Overview

The Wave Editor is the software which acquires wave data, then displays and edits them in AIFF and SD2 format using SCSP connected to SCSI I/F of Macintosh.
The Wave Editor has following functions:

- File operation
  Input/Output files, manage(such as save) and compare files and edit (such as mix) a file.
- Editing
  Edit (such as redo, cut and paste) files.
- Effect process
  Perform effect process on the current editing wave.
- SCSP process
  Input/Output sound for SCSP.
- TMP process
  Play the wave sound which is in the temporarily area, or revert it.
- Preference display
  Display SCSID.
- SCSI setting
  Perform SCSI process for Macintosh when wave data are input/output to/from Macintosh via SCSI from SCSP.

2.0 Description of Terminology

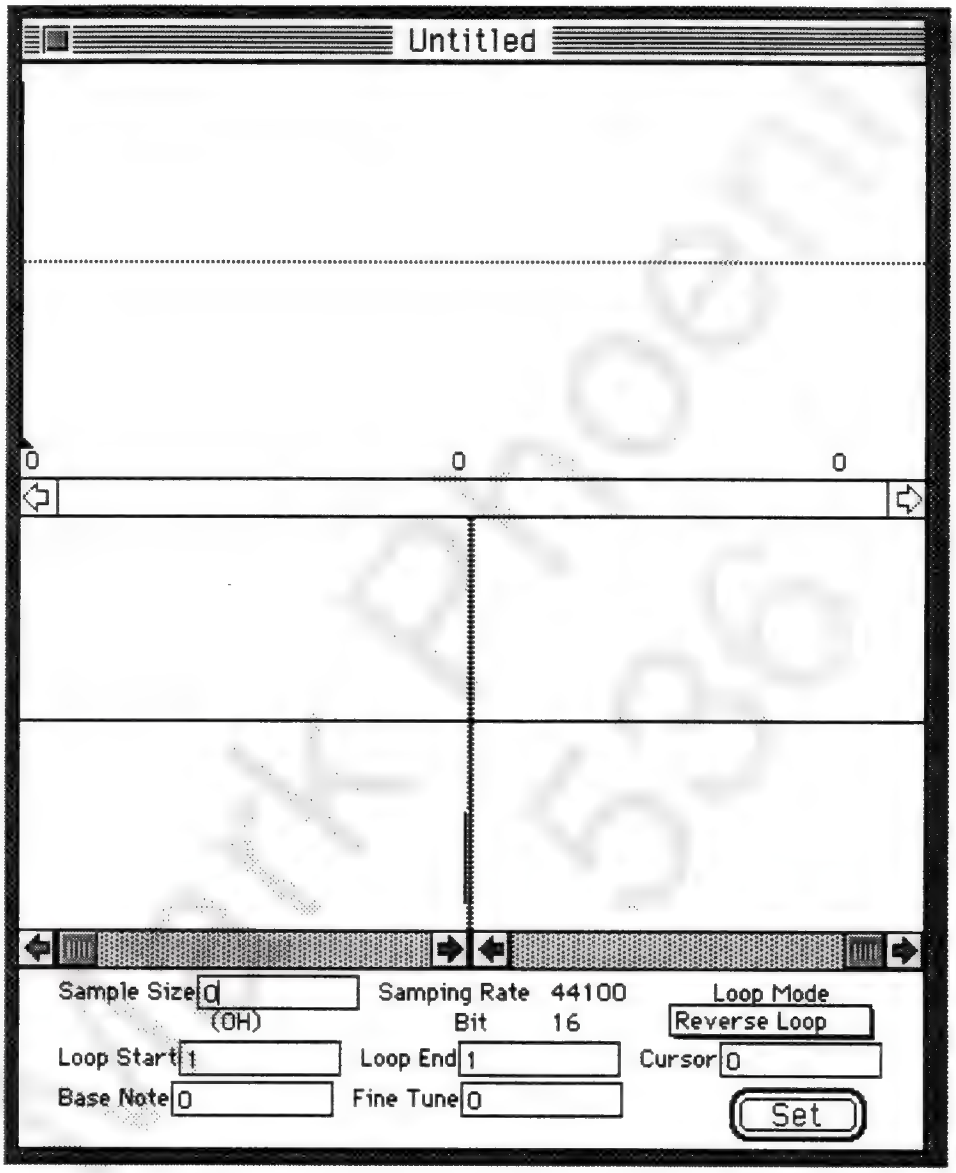
- AIFF file
  This Machintosh file includes loop information, Base Note and Fine tune information in addition to PCM data.
- Number of samples
  This is the number of words. For 8-bit, number of bytes; for 16-bit, half number of bytes is the number of samples.

3.0 Tutorial

A brief description on how to use this system is given here.

- Displaying the edit window
    1. Start the wave editor.
       Wave editor will first check which SCSP is loaded on, and if the board is installed on the Macintosh. It will exit if the board is not mounted.

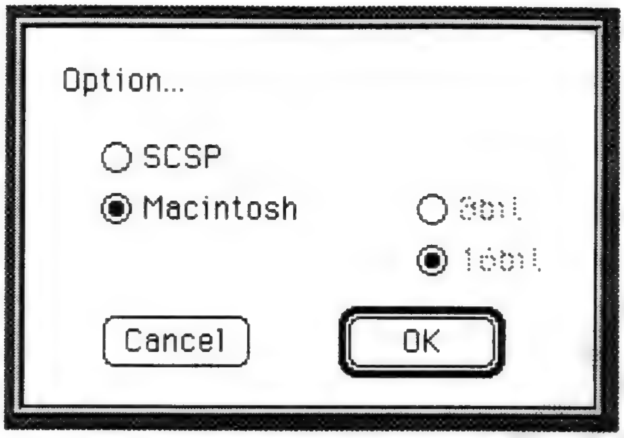
    2. Select "New" in "File" menu.
       The edit window for wave editing is displayed.



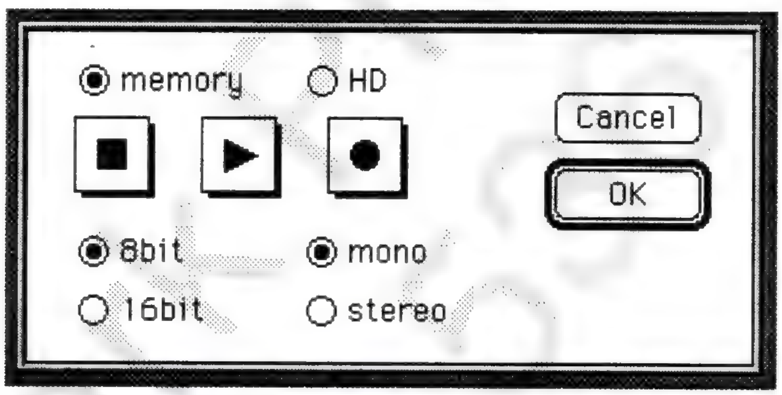
- Sound capture

  There is not any wave to edit yet. Let's capture the sound from SCSP board.

  1. Check that the source equipment is connected in front of digital-in of SCSP board.
  2. Select **Option...** in SCSP menu.
     The screen for setting the input and output equipment is displayed.

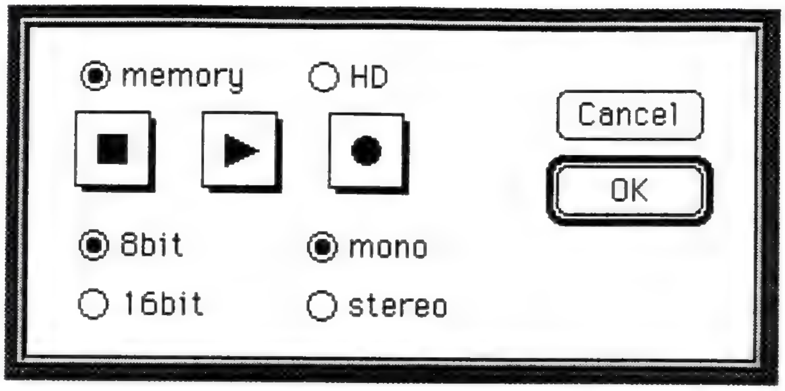


  3. Select SCSP.
     8-bit and 16-bit radio buttons are displayed.



  4. Set output to either 8-bit or 16-bit.
  5. Click on the **OK** button.
     System setting is done.

6. Select **Get Sound** in SCSP menu.
The window to capture sound is displayed.



7. Select **Memory** since short sound will be used.
8. Press recording button (i.e. black round button) after selecting 8-bit/16-bit and mono/stereo. Sound begins to be recorded as soon as the button is pressed. Recording will stop once the Stop button (Black square button) is pressed. If the **OK** button is clicked, it will return to wave edit window. At this time, the latest sound captured is displayed as a wave.

• Wave editing

Next, add a loop to this wave. There are two bars displayed near the wave used for a loop. Move the bar by dragging it. While dragging, the points of loop, which are displayed as number at the bottom, should be changing. Furthermore, the wave, which is displayed near the central loop, should also be changing. Next, let's move the loop edit slider. While moving, the bar and numbers for wave edit loop should be changing.

There are also other functions, such as cut and copy wave, filter, mix with other waves. Those functions will be described later.

4.0 Files to Use

The files used in Wave editor are as follows:

- Wave Edit data file
  This is the AIFF format created by the save process.
- Alchemy file
  This is the file created by Alchemy in AIFF format.
- Sound designer file
  This is the file created by sound designer in AIFF format.

5.0 Overview of Function

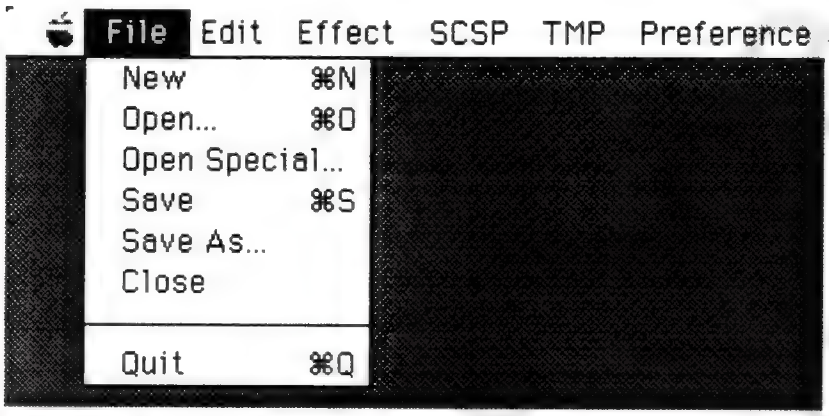
Menu bar and pulldown menu in wave editor and control window are introduced here.

Menu Bar

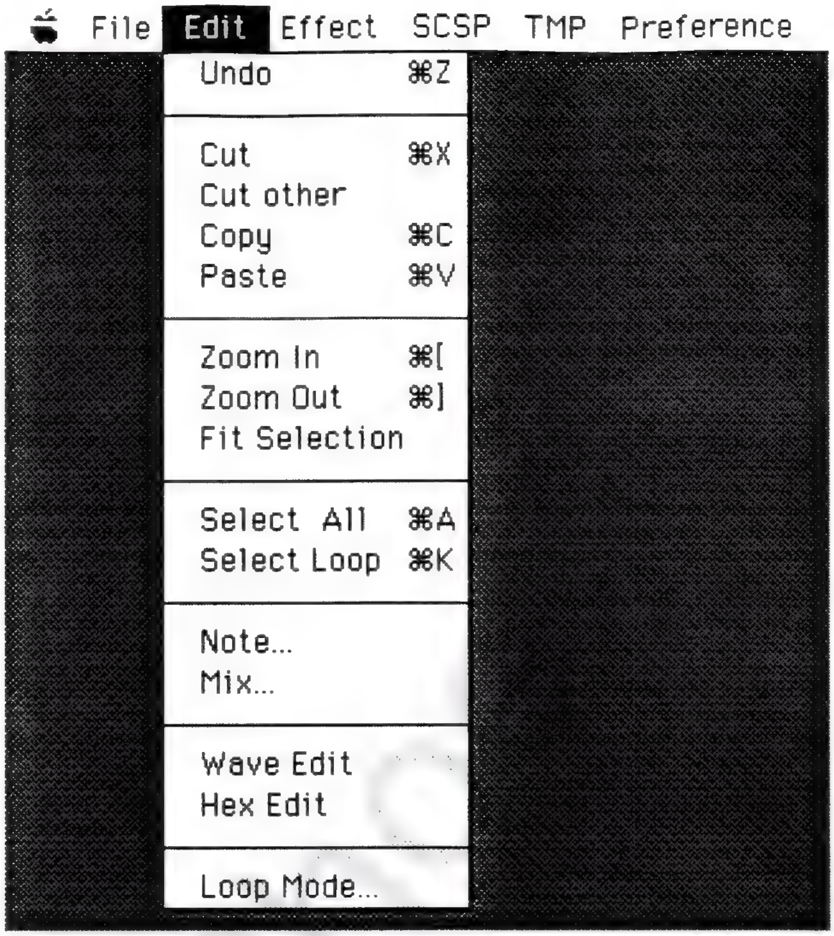
Pulldown Menu

- Apple menu
  This is the general Apple menu.

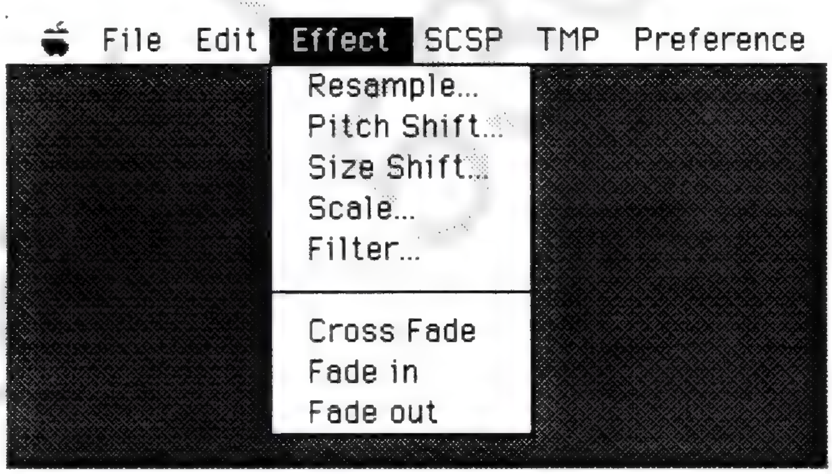
- File menu



- Edit menu

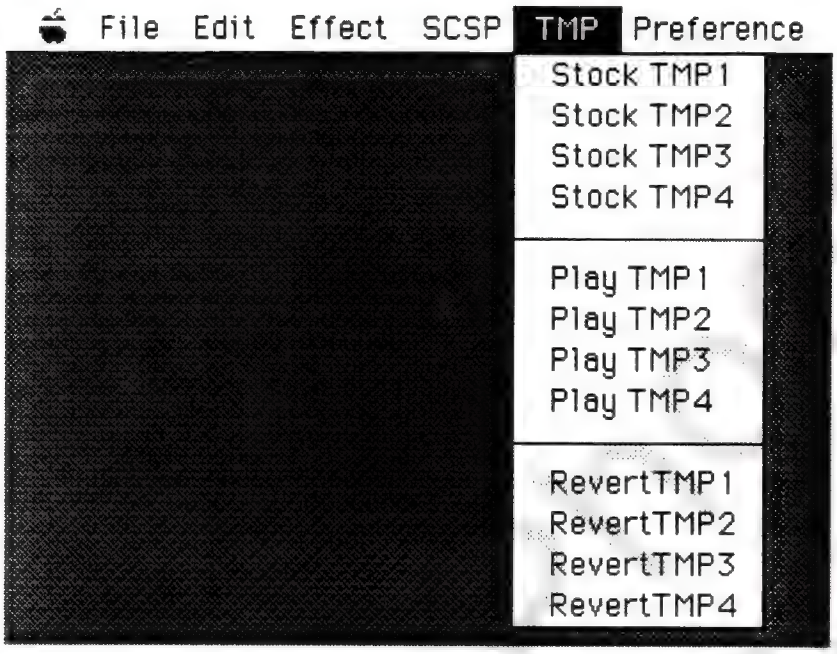


- Effect menu



• SCSP menu

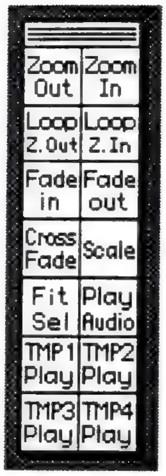
• TMP menu



• Preference menu

Control Window

The Control Window is displayed on screen at all times.
Clicking its icon is the same as selecting a menu. See page 39 for the functions of each icons.



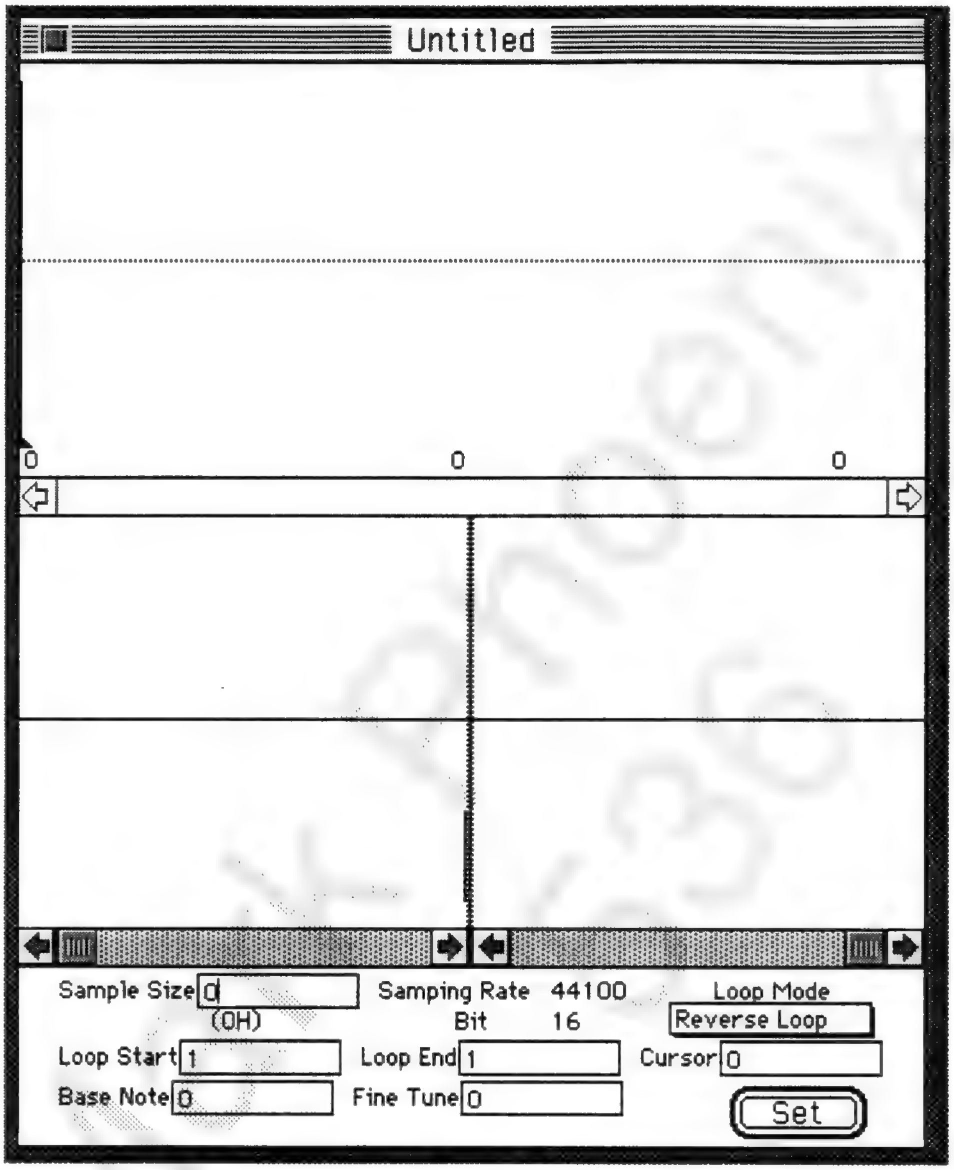
6.0 Details of Functions

Each menu item, and the function of each icon in the control window, is described here.

File Menu

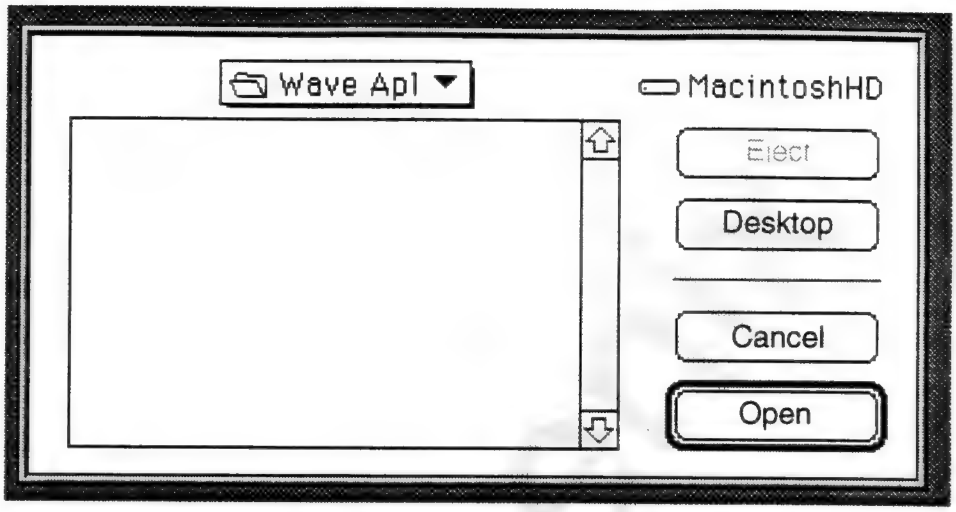
- New

  When PCM data needs to be captured from SCSP, open an edit window where wave is not captured.



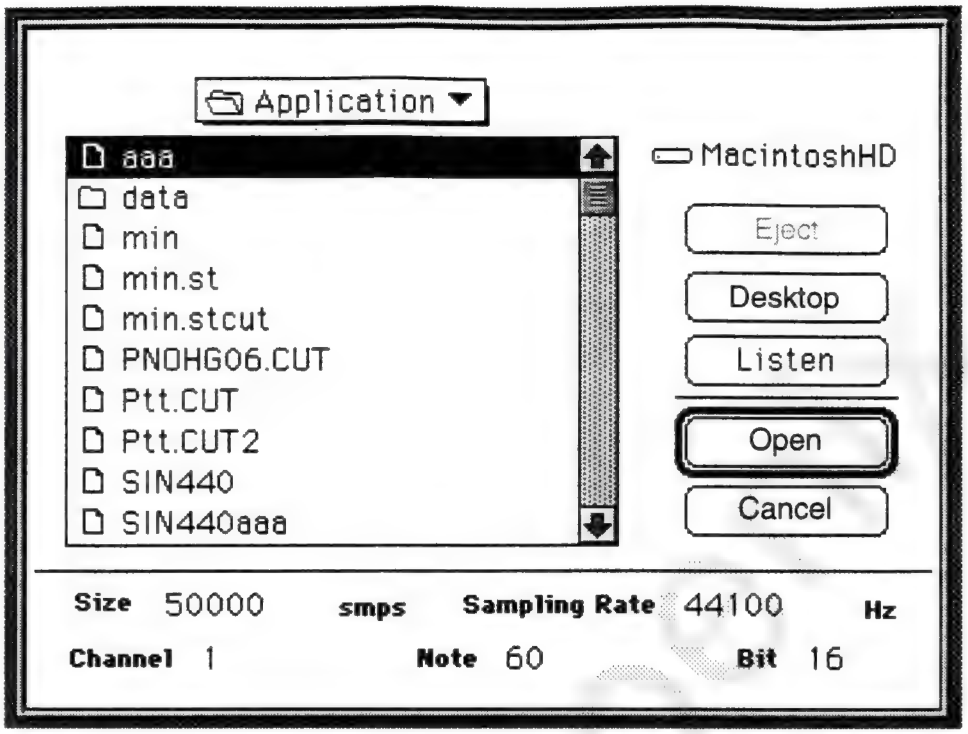
Select **Get Sound** in SCSP menu to display the wave on screen. Data will be captured from SCSP if **Start** is clicked, and will be completed if **Stop** button is clicked. The wave will then be displayed on screen.

• Open
  Open the saved file. By selecting this item, the window that displays the saved files (AIFF files only) will be displayed. Select the target file. The edit window capturing this wave will be opened by selecting the file.



For stereo, L, R window is displayed.

- Open Special...
  Select the file to open from the current window. A sound is heard just before opening the saved file. Information about the file is also displayed.



The parameters used in the "Open Special" window are as follows:

Listen
Listen to highlighted sound.

OK
Open selected file.

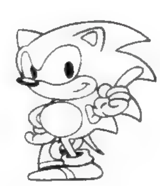
Size
Displays the sampling number of data.

Sampling rate
Displays sampling frequency.

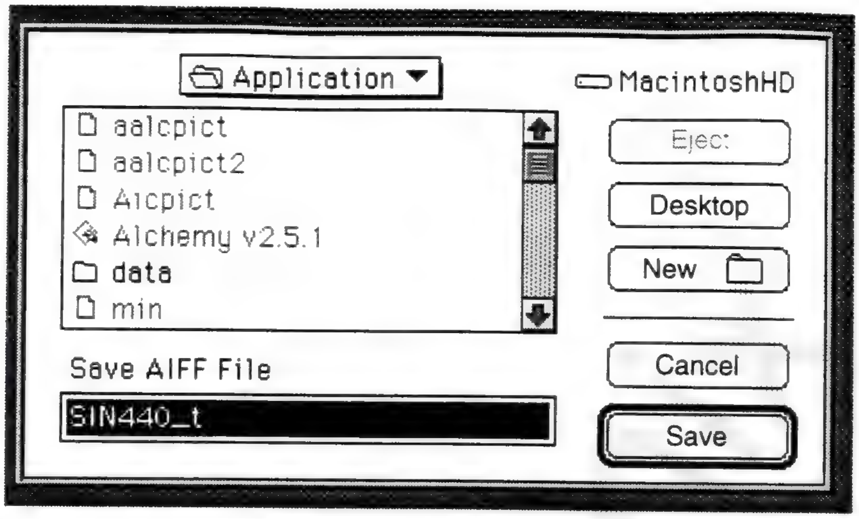
Channel
Display snumber of channels.

Note
Displays the frequency with MIDI note number.

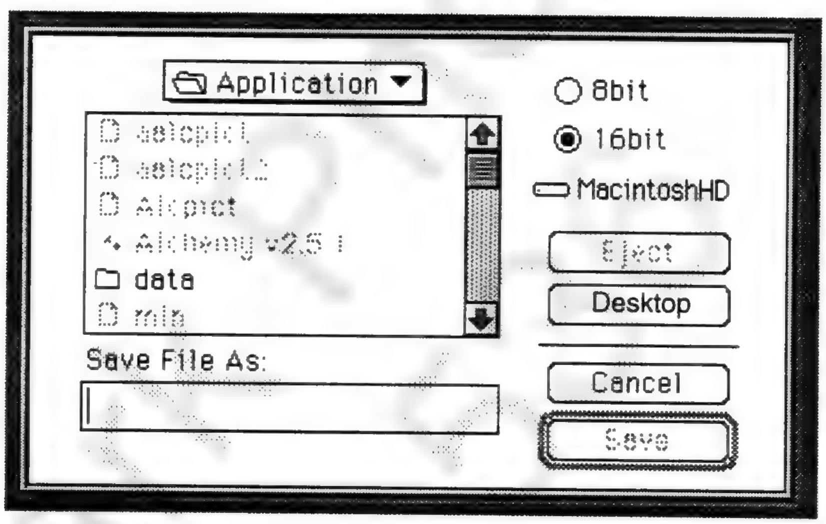
Bit
Displays the number of wave data bits.



• Save
  Saves current working file as SCSP data file. The following "Save" screen is displayed.



• Save As
  Saves current working file as SCSP data file with different file name. The following screen is displayed.

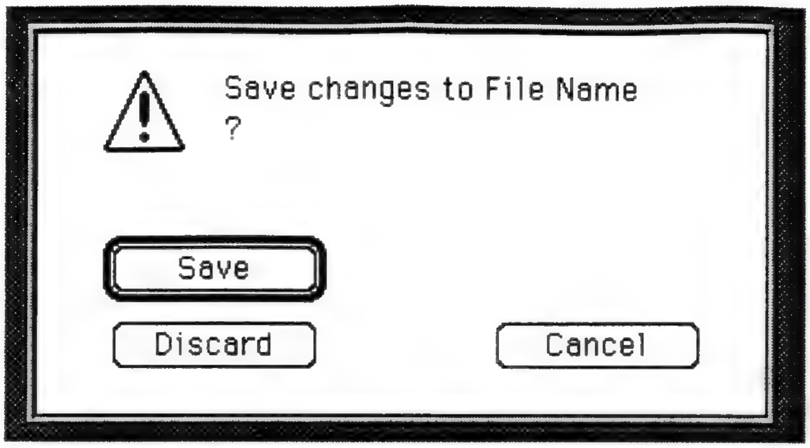


The parameters set in this window are as follows:

16-bit
Save in 16-bit Audio IFF format.

8-bit
Save in 8-bit Audio IFF format.

• Close
Closes current working file.

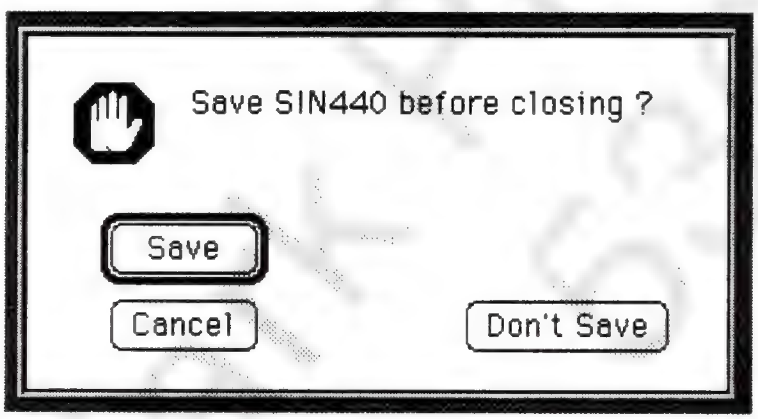


The buttons used in this window are as follows:

OK
Closes the window in its current state.

Save
Saves the file.

Cancel
Returns to Edit window. An alert window will appear if closing is attempted without saving changes.



The buttons used in Warning window are as follows:

Cancel
Cancels closing the window.

Don't Save
Closes the window without saving it.

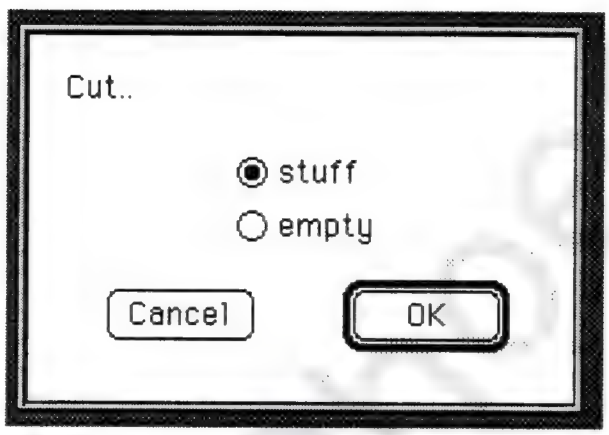
Save
Close the window after saving.

- Quit
  Quits wave editor and return to Finder.

Edit Menu

- Undo
  Cancels the execution of the latest operation.

- Cut
  The selection is cut and saved in the Clipboard. The following screen is displayed.

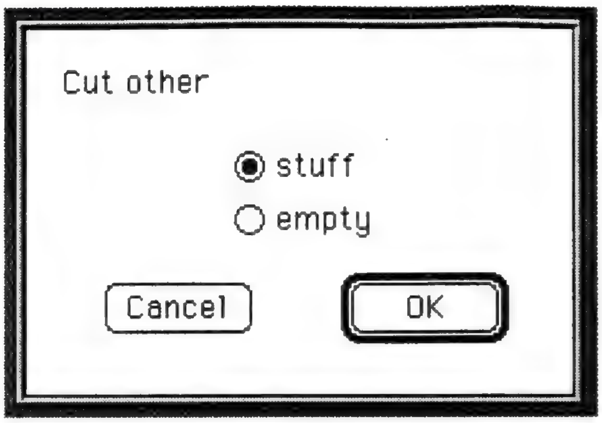


The parameters set in this window are as follows:

stuff
The space after cutting will be stuffed.

empty
The space after cutting will not be stuffed.

- Cut other
  The selected section is cut, but not saved in Clipboard.

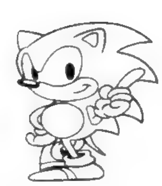


The parameters set in this window are as follows:

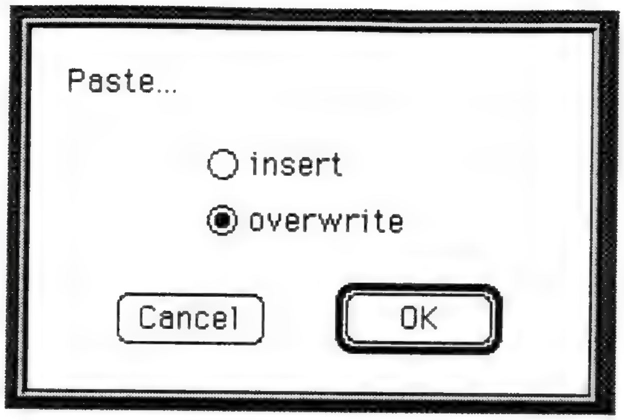
stuff
The space after cutting will be stuffed.

empty
The space after cutting will not be stuffed.

- Copy
  The selected section is copied, and saved in the Clipboard.



- Paste
  The image on Clipboard is pasted on current image.
  The following window will be displayed:



The parameters set in this window are as follows:

insert
Data is inserted on cursor.

overwrite
Data is overwritten on cursor.

- Zoom In
  Increases the displaying ratio of wave.
  It can be expanded to maximum 32768 multiples.

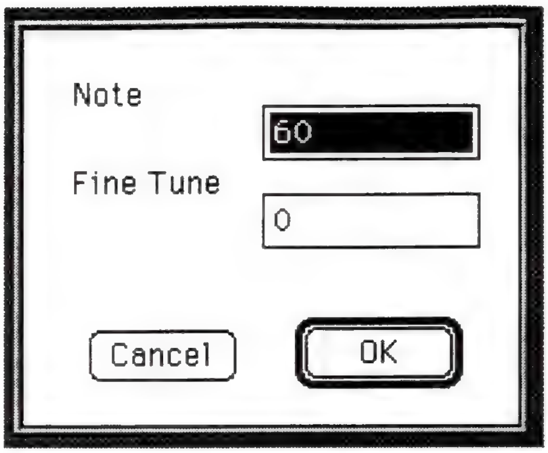
- Zoom Out
  Decreases the displaying ratio of wave.
  The minimum size is the one displayed when it is opened.

- Fit Selection
  Zooms in the selected section.

- Select All
  Selects the entire wave currently in the Edit window.

- Select Loop
  Only the loop portion of the wave in Edit window is selected.

- Note
  Set Note and Fine Tune. The following window is displayed.



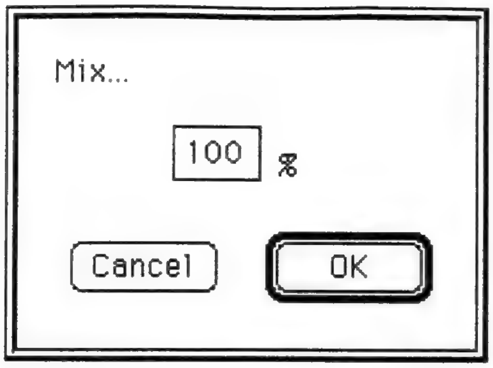
The parameters set in this window are as follows:

Note
Sets Note. Values of 0~127 can be set.

Fine Tune
Sets Fine Tune. Values of 0~127 can be set.

• Mix

Mixes the wave in Clipboard with selected part of current editing wave data. The following screen is displayed when "Mix" is selected.

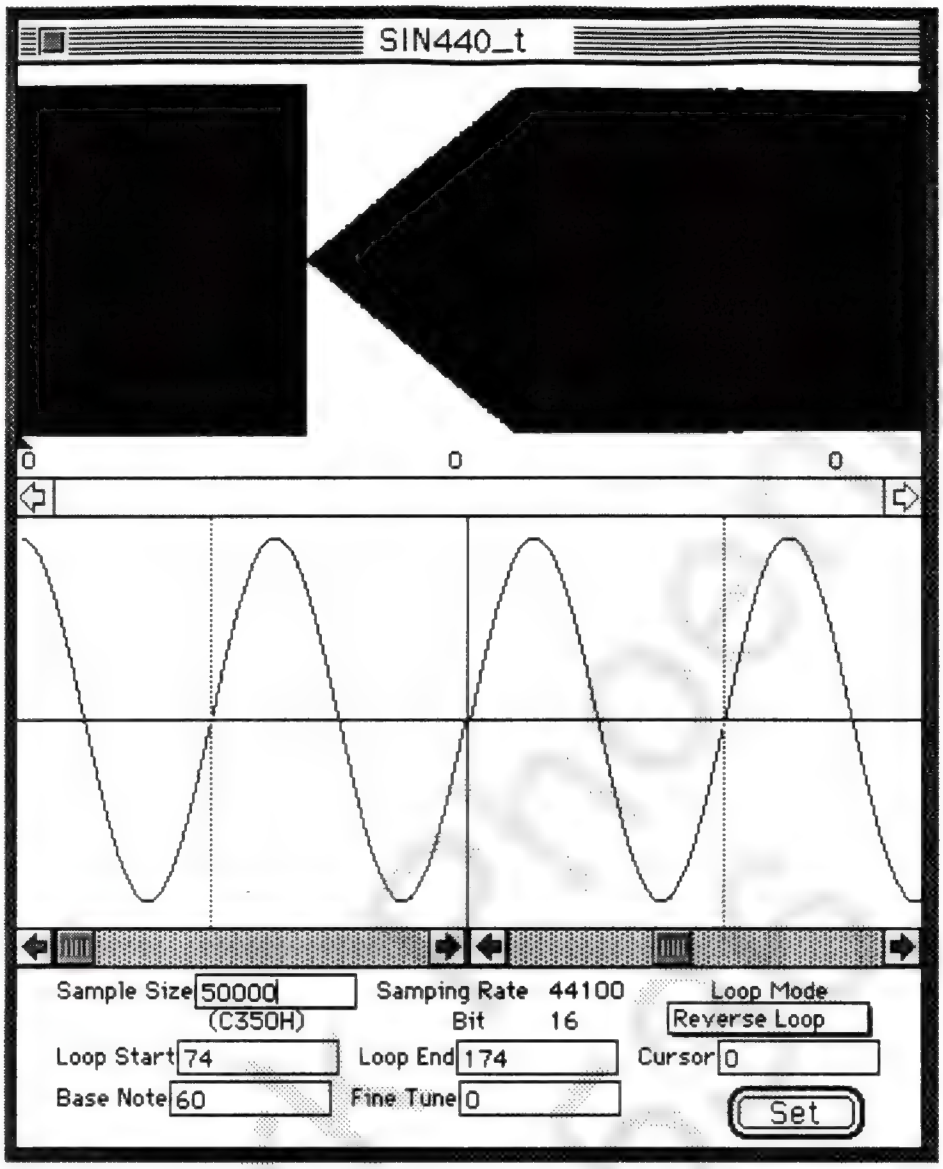


The parameters set in this window are as follows:

Mix
For the currently active wave, input the percentage of the mixed file which is currently selected. Values of 1~100 can be set.

OK
Converts the temporary data to mixed data.

- Wave Edit
  Turns the Edit window into the wave display screen.



There are three sections in wave display screen. The top is Wave Edit screen, the middle is Loop Edit screen, and the bottom is header display screen. These screens are described on the next page.

Wave Edit Screen

If there is still enough memory, this window can be opened repeatedly once **Open** is selected. Wave data opened will be displayed in this window when wave data from SCSP is displayed in "New" screen.

- The start and end point of a Loop is set in the screen by moving mouse cursor.
- The selected range can be displayed as highlighted by dragging the wave. When moving out of selected range, the displayed section will scroll itself.
- Clicking in the zoom box will expand the display of a window (and its contents).
- If you drag size box to expand, display of window and wave, they will be expanded in mouse moving direction. If you drag size box to reduce, only window will reduces its size while the display size of wave will not change in horizontal direction, and both display size of wave and window will reduce in vertical direction.
- When moving a loop point, the Loop Edit and header display screens will change.

Loop Edit Screen

- Left point is endpoint of loop while right point is start point of loop.
- If you move loop point, Loop Edit screen and header display screen will change.

Header Display Screen

Each item in Header display screen is described below.

Sample Size
Sets the sampling number of data.

Sampling rate
Displays sampling frequency.

Loop Mode
Determines loop mode when sound is played on SCSP or Macintosh.

Forward Loop
Plays in normal direction.

Reverse Loop
Reads the wave data from reverse direction to play.

Alternate Loop
Reads the loop in normal or reverse directions alternatively.
This is effective only when SCSP is selected in "Option". When Macintosh is selected, play is always Forward Loop even when Alternate Loop is selected.

Loop off
Loop is not run.

Bit
Displays the bit number of wave data.

Loop Start
Displays and sets the start point of loop.

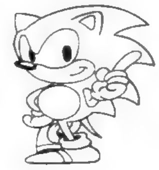
Loop End
Displays and sets the endpoint of loop.

Cursor
Displays and sets cursor location.

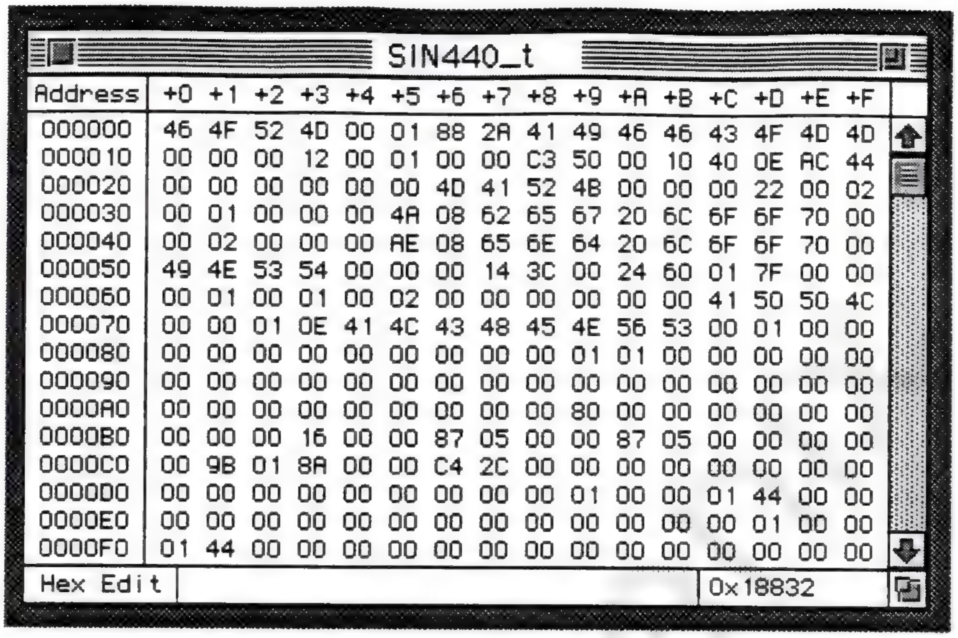
Base Note
Sets Note. Values of 0~127 can be set.

Fine Tune
Sets Fine Tune. Values of 0~127 can be set.

Set
Reflects the header to wave display.



- Hex Edit
  Sets Edit window to Hex decimal display screen.



The functions used in this window are as follows.

Cursor
- The data in currently selected cursor is displayed in reverse.
- Scrolls automatically when mouse is dragged towards outside of list display area.

Keyboard
For data in the cursor, values of 0~9 and A~F can be input.

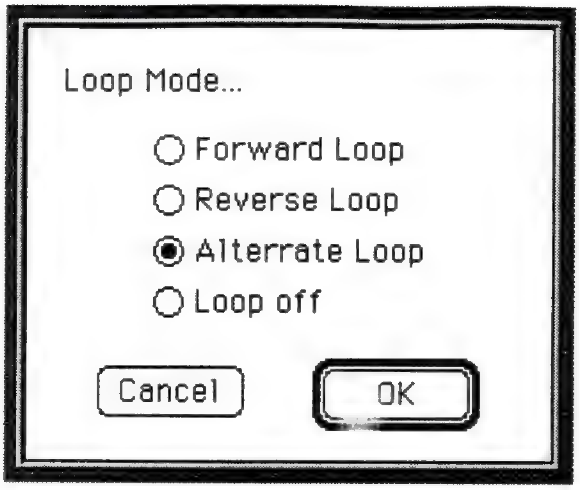
Display
Data of Loop Start address and Loop End address are displayed in bold.

Scroll bar
Works the same way as the regular scroll bar.

Used RAM
Displays wave size in hex decimal number.

Window
- File name is displayed in Title bar.
- Size box and scroll bar are the same as described in standard specification.

- Loop Mode
  Determines loop mode when sound is played on SCSP or Macintosh. The following screen is displayed.



The parameters set in this window are as follows:

Forward Loop
Plays normally.

Reverse Loop
Reads the wave data from reverse direction to play.

Alternate Loop
Reads the loop in normally or reversed alternatively.
This is effective only when SCSP is selected by "Option". When Macintosh is selected, play is always Forward Loop even if Alternate Loop is selected.

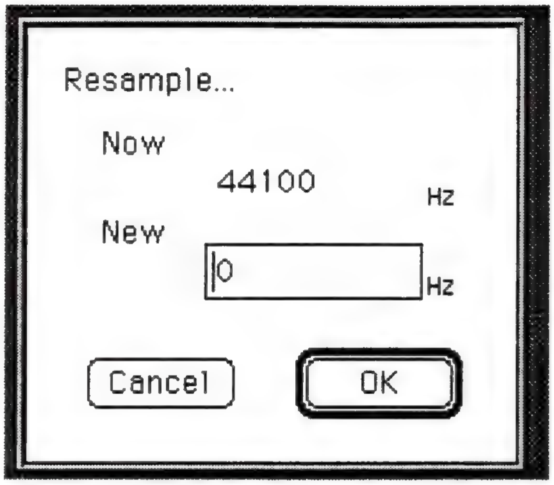
Loop off
Loop is not run.

OK
Changes the loop mode of SCSP or temporary memory to specified mode.

Effect Menu

Each item in Effect menu is described here.

- Resample
  Resampling the current editing wave. The following window is diplayed:



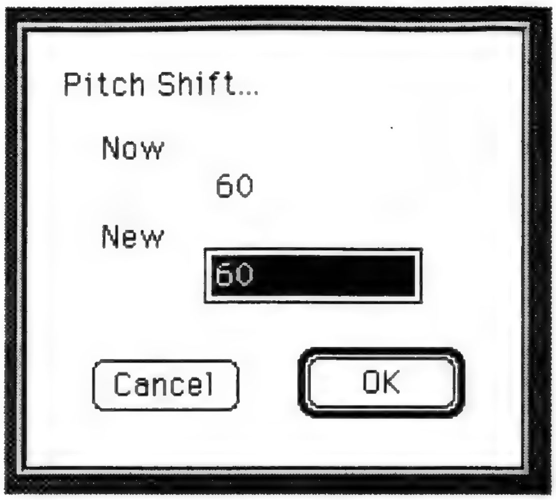
The parameters set in this window are as follows.

Now
Displays the current sampling frequency.

New
Sets new sampling frequency. Values of 1~65535 can be set.

OK
Re-writes data of temporary memory in AIFF file.

- Pitch Shift
  Run "Pitch Shift" on current editing wave. The following window is displayed.



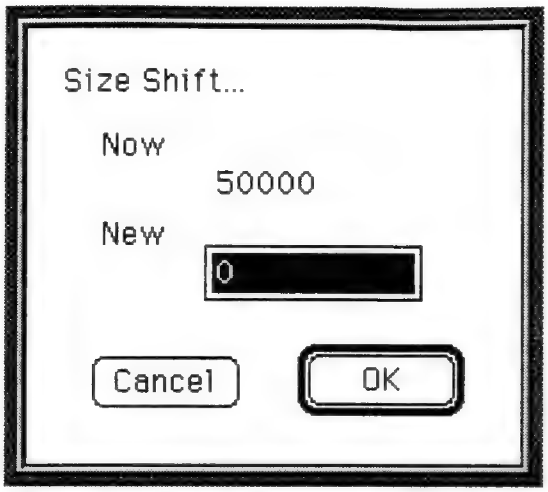
The parameters set in this window are as follows.

Now
Displays current frequency (MIDI code).

New
Sets new frequency (MIDI code). Values of 0~127 can be set.

OK
Re-writes data of temporary memory in AIFF file.

- Size Shift
  Re-sizes the sample of current editing wave. The following window is displayed.



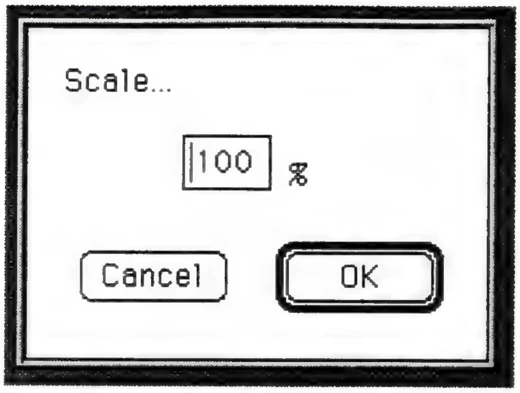
The parameters set in this window are as follows.

Now
Displays the current sample size.

New
Sets new sample size. Values of 0~4294967295 can be set.

OK
Re-writes data of temporary memory in AIFF file.

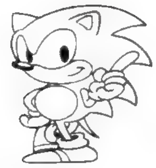
- Scale
  Re-scales the amplitude of currently editing wave. The following window is displayed.



The parameters set in this window are as follows:

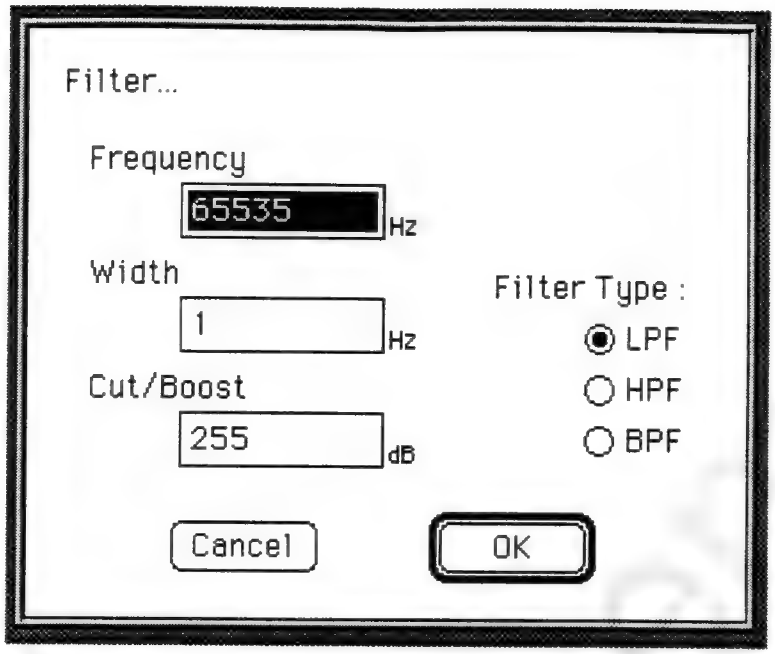
Rate
Inputs the percentage for scaling. Values of 1~200 can be set.

OK
Re-writes data of temporary memory in AIFF file.



- Filter

  Runs filter on current editing wave. The following window is displayed.



The parameters set in this window are as follows.

Frequency
Input Hz number to determine lowest frequency to be cut.
Values of 1~65535 can be set.

Width
Input in Hz the frequency below which other frequencies must be cut.
Values of 1~65535 can be set. This is effective only when the Filter Type is BPF.
Others will be displayed in gray.

Cut/Boost
Input dB number to determine which frequency band to cut. Values of 1~255 can be set.

Filter Type
Select filter to set in the LPF, HPF or BPF radio button.

- Comp
  Runs compressor on current editing wave.

- Noise Gate
  Runs Noise Gate on current editing wave.

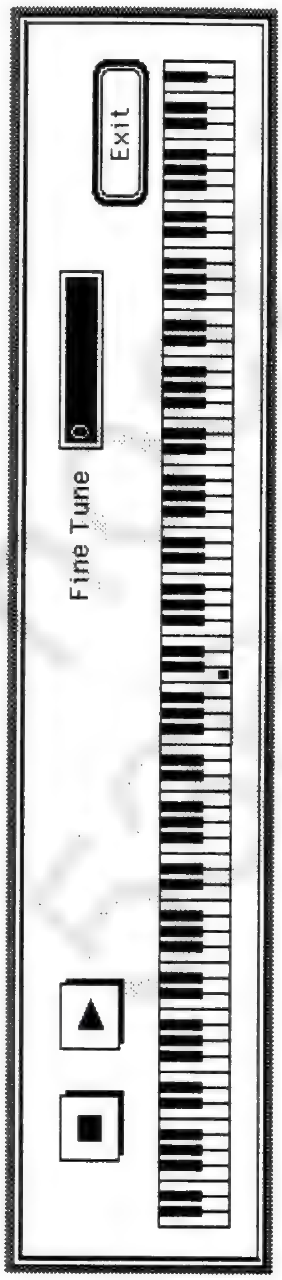
- Cross Fade
  Runs Cross-fade on selected wave with the wave in Clipboard.
  At the same time, re-writes the data of temporary memory in AIFF file.

- Fade in
  Runs Fade in on current selected section of wave. At the same time, re-write the data of temporary memory in AIFF file.

- Fade out
  Runs Fade out on current selected section of wave. At the same time, re-write the data of temporary memory in AIFF file.

SCSP Menu

- Play Audio
  When the wave range is specified for output destination selected in "Option", sound is played for specified portion of the wave. When the wave range is not specified, sound is played for the whole wave. The following window is displayed when this item is selected.



The icons used and the parameters set in this windows are as follows:

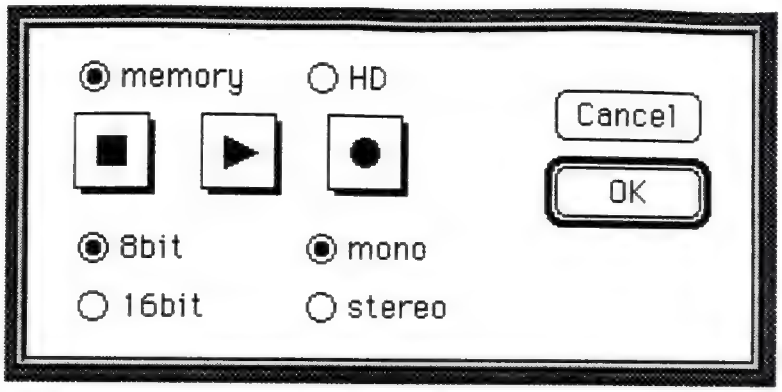
Stop
Stops playing.

Play
Starts playing.

Fine Tune
Sets Fine Tune. Values of 0~127 can be set.

Keyboard
The sample note can be played by clicking the keyboard.

• Get Sound
  Captures sound from the SCSP, and displays a wave on the currently active Edit window.



The parameters set in this window are as follows.

Stop
Stops playing and recording.

Play
Sounds SCSP.

Rec
Starts recording.

Memory
Sets to memory recording mode.

HD
Sets to hard disk recording mode.

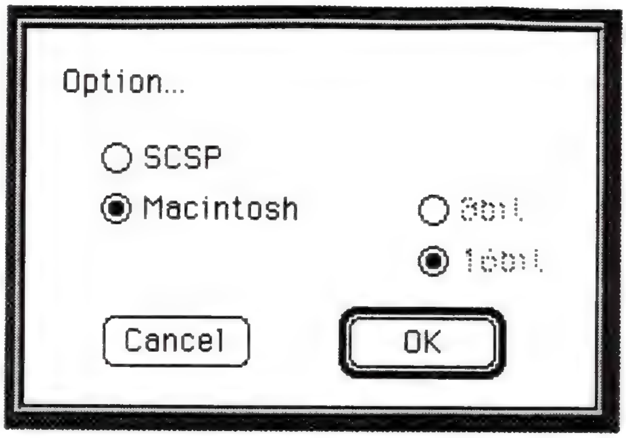
8-bit
Captures via 8-bit PCM data.

16-bit
Captures via 16-bit PCM data.

Mono
Input/Outputs in mono mode.

Stereo
Input/Outputs in stereo mode.

- Option
  Sets output destination of Play Audio and bit numbers of output. Menu can be selected even if file is not yet opened. The following window is displayed.



The parameters set in this window are as follows.

Audio Output
Selects output destination of Play Audio, and Macintosh or SCSP by radio button.

Output Bit
Selects bit numbers of output. This is effective only when SCSP is selected by Audio Output. Select 8-bit or 16-bit by radio button.

TMP Menu

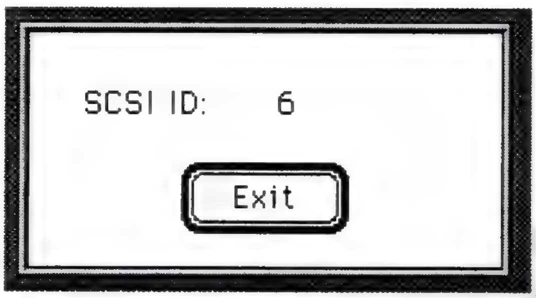
- Stock TMP
  Creates a temporary file for the wave which is currently active.

- Play TMP
  Plays sound for temporary file. Becomes Enable status only when temporary file is created.

- Revert TMP
  Reverts data saved in TMP in current active window.

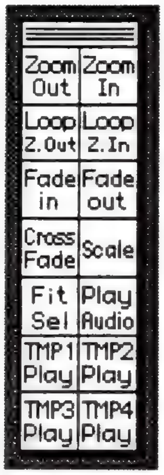
Preference Menu

- SCSI info
  Displays current SCSI ID.



Control Window

Each icon in the Control window is described here.



- Loop Zoom Out
  Reduces Loop Edit screen.

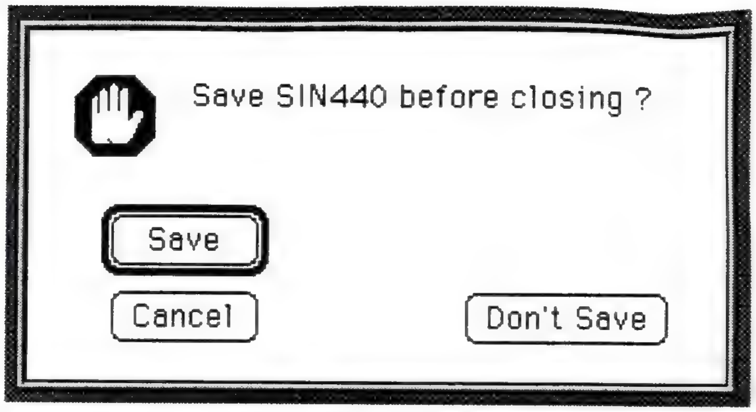
- Loop Zoom In
  Expands Loop Edit screen.

- Fit Selection
  Zooms in on selection.

- Other icons
  Other icons works the same way as when menu bar is selected.

7.0 Error Process

If there are any errors, the operation will be stopped and error dialog box displayed.



Description of communication errors with SCSP and other application errors will be displayed in this dialog box.

SEGA

SEGA OF AMERICA, INC.
Consumer Products Division

Standard MIDI File:
Saturn
Converter Specification

Doc. #ST-66-121593

NOTICE

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.

2.

3. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

4. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

5. No license is granted by implication or otherwise under any patents or other rights of any third party or SEGA Enterprises, Ltd.

6. This document is confidential. By accepting this document you acknowledge that you are bounded by the terms set forth in the non-disclosure and confidentiality agreement signed separately and/in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

(4/20/94 - 001)

October 7, 1993
Katsutaka Nitta, Sound Development Section, Software Engineering Department

1.0 Change History

September 20, 1993 Ver. 1.1

- Added a fade compression mode for pitch bend and changed the contents of the header flag accordingly.
- Changed the tempo track from 8-byte units to 6-byte units.
- Added a command used in the regular tracks and provided the Extend Gate (Step) Time command in stages for each amount of extension.

October 4, 1993 Ver. 1.2

- Of the commands used in the regular tracks, the Jump, Call, and Return commands were eliminated and the Reference command provided in their place. The Loop Start and Reserve commands were added.
- The allocation of command numbers was changed to reflect a statistical classification.
- Changed the Step Time abbreviation rule for Note On and various events. Until now, Step Time was abbreviated only when Note On was the same as the preceding Note On and Step Time, but this has been changed to the preceding event and is not limited to Note On. Further, if the preceding event and Step Time are the same for other events, as well, Step Time is omitted and the 7th bit of the data byte is set to ON.
- Technical explanations necessary for data restoration have been provided where needed.

October 7, 1993 Ver. 1.21

- The abbreviation rules for the top byte of events (command number) and the Step Time have been clarified.
- Typographical errors were corrected.

November 1, 1993, Ver. 1.3

- Music piece = file size (bytes 4 and 5) was inserted in the output file header.

2.0 Preference Settings

The operation of this converter is determined by the configuration (preference setting) program M6CNF.EXE, which exists separately from the converter. The following operations can be set.

- Whether a temporary file is output or not.
- Whether ASCII format or binary format is used for output files.
- Output events other than those to be converted (meta events other than tempo) as comments. (This is valid only when the output file is in ASCII format. Also, comments are output only in temporary files.)
- Allow pitch bend to fade before saving, or save it without fading.

These are all Y or N switches; when M6CNF.EXE is started, they are opened {Translator's Note: Due to copy quality, it was difficult to distinguish "opened." It may be "heard."} in order and the responses are saved in the above order as Y or N in a text file called M6CNV.CNF. The converter reads these text files, which determine its operation.

The converter outputs both temporary files and complete files. Their differences are described below.

- Temporary files (extension: .TMP; Macintosh files are truncated after the 8th character)

  Output only when output is allowed by the configuration program.
  Output only when output is displayed as ASCII format and it has been instructed that events other than those to be converted to be output as comments, events other than those to be converted will be output as comments.
- Complete files (extension: .CNV; Macintosh files are truncated after the 8th character)

  These are always output regardless of what is set by the configuration program. They do not include events other than those to be converted, which are not even output as comments. Repeated detection is already completed. Therefore the Reference command is included. (If a repeat is not found, then of course the Reference command is not included.)

The configuration program does currently exist, but it is planned to eliminate it and include it as one window in the converter program to facilitate setting by way of a menu.

3.0 Events to Be Converted

Of the events included in a standard MIDI file, those that are converted by the converter are listed in the following table. These are converted with no fading as long as saving with fade of Pitch Bend has not been specified in the preference settings.

Event Description	Status Before Conversion	Conversion	Remarks
Note On/Off	9nH , 8nH	Yes	Note Off is replaced by Gate Time.
Poly-Key Pressure	AnH	Yes	
Control Change	BnH	Yes	
Program Change	CnH	Yes	Must exist at the top of each track.
Channel Pressure	DnH	Yes	
Pitch Wheel Change	EnH	Yes	Expressed with 7 bits and 14 bits
System Message	F0H - FEH	No	Exclusive, or Start, Stop, Song Position, etc.
Meta Event	FFH	Yes	Only tempo.

Meta Event is valid only for items for which tempo has meaning and is included in the tempo track. Also, in regular tracks no meta event undergoes conversion.

4.0 When an Error Is Output

When the following conditions are not satisfied, the converter outputs an error message and stops the conversion operation.

- The standard MIDI file prior to conversion does not include a system message. System messages include Song Position Pointer, System Exclusive, Song Select, etc.

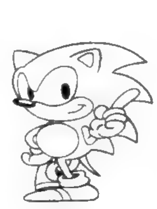
- The number of events included in a standard MIDI file prior to conversion is less than 6143 events per track. However, a slightly smaller number of events is output after conversion (this is because Note Off is absorbed in Gate Time of Note On). Similarly, the number of meta events is less than 256, and the length of 1 meta event cannot exceed 127 bytes.

- The standard MIDI file must be of type #1. Future versions will be compatible with type #0 files, but compatibility with type #2 files is not planned.

- The preference setting file (M6CNV.CNF) and the standard MIDI file to be converted must be in the same directory as the converter program (M6CNV.EXE).This is a requirement left over from when the MS-DOS converter and the source files were used in common. This will be changed in the future.

- No more than one Loop Start command (31st Control Change) can exist in each track.

- A Program Change must be present at the top of each track. However, empty tracks do not require a Program Change.



Structure of Output Files

Offset from top	Description	Number of bytes	Notes
0	Resolution	2	
2H	Number of tracks used	1	
3H	Header flag	1	
4H	Number of bytes of this tune	2	File size
6H	(Reserved)	2	For future expansion
8H	Tempo track offset address	2	
0AH	Track #1 offset address	2	
Undefined	Track #2…	2	
Undefined	…	2	
Undefined	Track #n offset address	2	
Undefined	Tempo track, tempo data	Undefined	
Undefined	Track #1, play data	Undefined	
Undefined	Track #2…	Undefined	
Undefined	…	Undefined	
Undefined	Track #n play data	Undefined	

Each is explained later in the next few pages.

Resolution

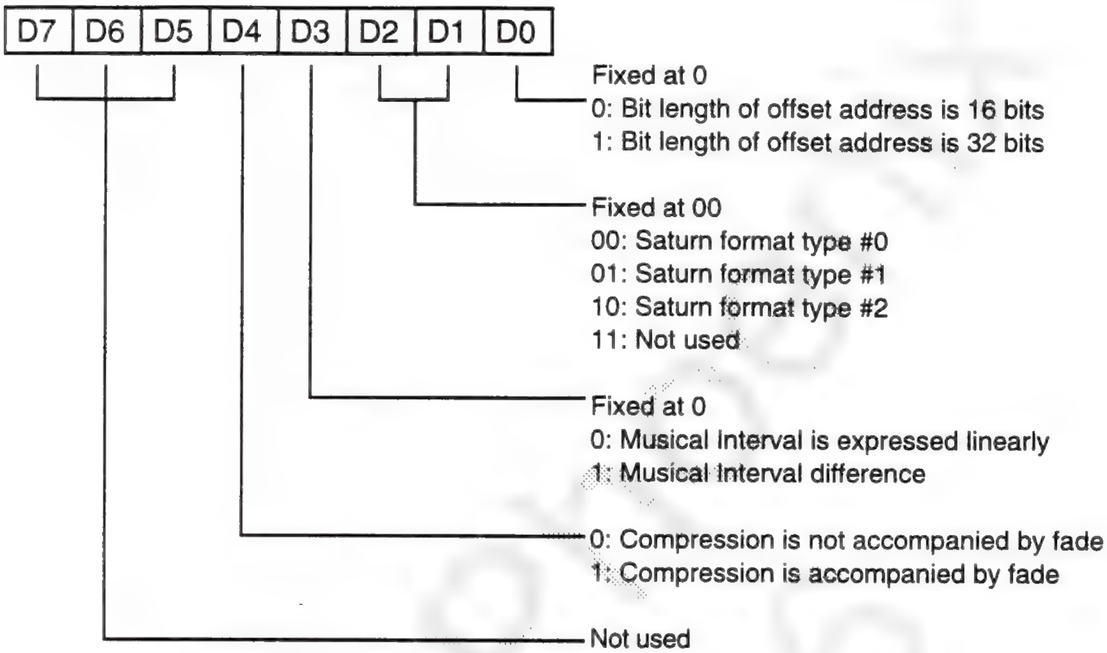
The resolution is the number of counts per quarter note of the original MIDI data. It is expressed with 2 bytes in most significant (MS) and least significant (LS) order.

Number of Tracks Used

The number of tracks used is the number of tracks of the Saturn format data after conversion, including the tempo track.

Header Flags

Header Flags indicate the attributes of the overall data.



D4 indicates whether or not the pitch bend data is faded during conversion. In the case of pitch bend, the change width is normally saved in 14 bits, but it can also be recorded using only the upper 7 bits. In that case, this D4 becomes "1."

Beacuse a pitch bend that has been faded and a pitch bend that has not been faded have different command numbers, the sound source driver does not need to look at D4 of the header flag to perform judgment.

Format of Tempo Tracks

Position	Description	
+0 byte	Upper byte of tempo value	
2	Lower byte	
3	Least significant byte	
4	Upper byte of count number	Duration until next tempo change
6	Lower byte	
7	Least significant byte	

Written in 6-byte units. The tempo value is expressed in the same way as in standard MIDI files, and indicates the number of microseconds per beat.

Because no command expressing the track end is inserted in the tempo track, the end must be determined by calculating the size of the tempo track from the number of offset bytes of the header.

There are items in the MIDI sequencer that do not output a tempo when the tempo (BPM) is 120 tunes. In this case the converter does not output the information tempo 120, and because the tempo track is output empty (size is 0 bytes), caution is required on the play side.

Loop and Tempo Changes

Loops must be specified independently for each track. In other words, a common loop cannot for all tracks be specified. Therefore the tempo cannot be changed in a loop.

In a standard MIDI file a loop command is not defined, but loops are required when actually composing. When defining a loop command, one common for all tracks and those independent for each track cannot exist together— for example, when the ranges of an overall loop and of track-independent loops overlap. Also, a complex loop may result in a different tune being played during play while composing on a MIDI sequencer and after conversion.

From the standpoint of effectiveness, track-independent loops are considered more versatile, and therefore loops common to all tracks have not been included.

Therefore, because tempo changes are not done independently for each track, the tempo cannot be changed as independent loops for each track are put together.

Format of Non-Tempo Tracks (Regular Tracks)

The following commands and events exist and are saved together in regular tracks. Those items in the original standard MIDI file are events, and Reserve, Extend Gate (Step) Time, and so on are commands (there is no clear distinction). These are explained in the following tables.

00 - 7F: Note On
80 - 8F: Control rests and play flow
90 - 9F: MIDI events other than Control Changes
A0 - AF: Make Control Changes and Control Changes used with high-frequency independent events
B0 - BF: Gate/Step extension

*Skipped numbers exist as other events in the conversion operation in the converter memory.

80H is Note Off, 90H is Note On, and 8FH is a deleted event (event omitted by the Reference command).

FFH is also skipped.

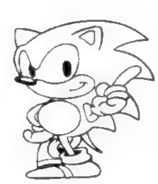
Top Byte	Description
Less than 7FH	Note On
Fixed at 80H	(Skipped, cannot be used)
81H	Rest
82H	Reserve
83H	Reference
84H	Loop Start
85H	(Not used)
86H	(Not used)
87H	(Not used)
88H	(Not used)
89H	(Not used)
8AH	(Not used)
8BH	(Not used)
8CH	(Not used)
8DH	(Not used)
8EH	End of Track
8FH	(Skipped, cannot be used)

Top Byte	Description
90H	(Skipped, cannot be used)
91H	Poly-Key Pressure
92H	Program Change
93H	Channel Pressure
94H	Pitch Bend (14-bit expression)
95H	Pitch Bend (7-bit expression)
96H	(Not used)
97H	(Not used)
98H	(Not used)
99H	(Not used)
9AH	(Not used)
9BH	(Not used)
9CH	(Not used)
9DH	(Not used)
9EH	(Not used)
9FH	(Not used)

Top Byte	Description
A0H	Control change
A1H	Modulation
A2H	Breath control
A3H	Foot control
A4H	Main volume
A5H	Panpot
A6H	Expression
A7H	(Not used)
A8H	(Not used)
A9H	(Not used)
AAH	(Not used)
ABH	(Not used)
ACH	(Not used)
ADH	(Not used)
AEH	(Not Used)
AFH	(Not Used)

Top Byte	Description
B0H	Extend Gate Time 200H
B1H	400H
B2H	600H
B3H	800H
B4H	A00H
B5H	C00H
B6H	E00H
B7H	1000H
B8H	Extend Step Time 100H
B9H	200H
BAH	400H
BBH	600H
BCH	800H
BDH	1000H
BEH	1800H
BFH	2000H

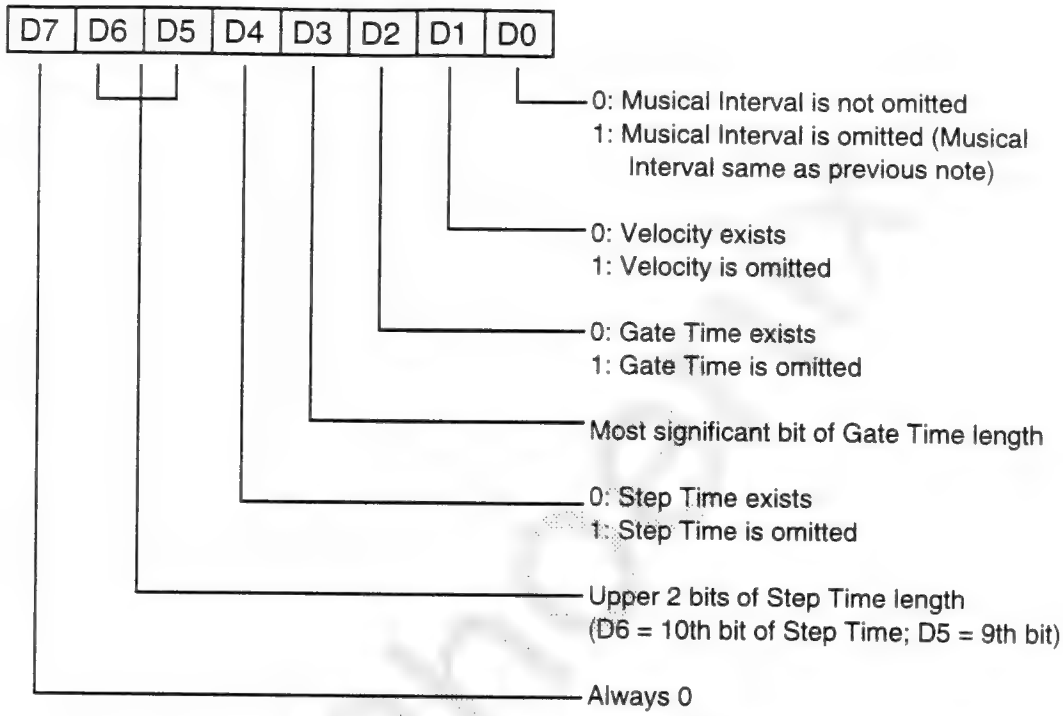
Top Byte	Description
C0H	(Not Used)
C1H	(Not Used)
C2H	(Not Used)
C3H	(Not Used)
C4H	(Not Used)
C5H	(Not Used)
C6H	(Not used)
C7H	(Not used)
C8H	(Not used)
C9H	(Not used)
CAH	(Not used)
CBH	(Not used)
CCH	(Not used)
CDH	(Not used)
CEH	(Not used)
CFH	(Not used)



00 - 7FH: Note On

Status (1 byte)	Interval (1 byte)	Velocity (1 byte)	Gate Time (1 byte)	Step Time (1 byte)

Status includes the status of subsequent data. Each bit has meaning.



The Musical Interval, Velocity, and Gate Time are omitted when they are the same as the previous note (meaning Note On, not the preceding event).

The Step Time is omitted when it is the same as the preceding event (not limited to the preceding Note On).

The Musical Interval and Velocity are expressed with 7 bits, as in MIDI.

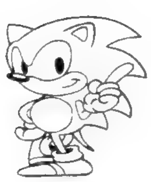
The Gate Time is 9 bits long to match the information in Status, and the Step Time is 10 bits long. If these are insufficient for certain note lengths, the Extend Gate (Step) Time command can be put in front of the Note on.

If Gate time and Step time are omitted, of course, the Extend gate (step) time command is not output. These are not omitted, however, if there is the same data as the preceding event at the top of an event group referenced by the Reference command in a complete file (file in which repetition is detected), and therefore Note On takes on a 5-byte length.

81H: Rest

81H (1 byte)	Step (1 byte)

If the time until the first event to be converted is not zero at the top of a track, this command is inserted at the top of the track. Because this command exists only at the top of a track, and no more than one exists in one track, the 2nd-byte Step Tme of this command is not omitted.



82H: Reserve

82H (1 byte)	Number of Repetitions (1 byte)

To avoid outputting an event each time the same event (except Note On) is repeated three or more times, this Reserve command is used to reduce the amount of data by reserving all in advance.

The Reserve command is used to omit the top byte of the second and subsequent events after some event is output without the top byte being omitted (it is possible that Step Time may be omitted). Some examples follow.

```
95H , 40H , 01H          ;Pitch Bend ( Value = 64)
82H , 06H                ;Reserve ( Reserved 4 events )
42H , 05H                ;Pitch Bend ( Value = 66 )
50H , 08H                ;Pitch Bend ( Value = 80 )
55H , 04H                ;Pitch Bend ( Value = 85 )
60H , 05H                ;Pitch Bend ( Value = 96 )
```

In principle, commands to be reserved are limited to those that appear repeatedly, like those shown in the folllowing table.

Top Byte	Description	Reserved		Top Byte	Description	Reserved
7FH or less	Note On	No		A0H	Control change	Yes
81H	Rest	No		A1H	Modulation	Yes
82H	Reserve	No		A2H	Breath control	Yes
83H	Reference	No		A3H	Foot control	Yes
84H	Loop Start	No		A4H	Main volume	Yes
8EH	End of Track	No		A5H	Panpot	Yes
91H	Poly-key Pressure	Yes		A6H	Expression	Yes
92H	Program Change	Yes		B0H-B7H	Extend Gate Time	No
93H	Channel Pressure	Yes		B8H-BFH	Extend Step Time	No
94H	Pitch Bend (14 bits)	Yes				
95H	Pitch Bend (7 bits)	Yes				

* The Extend Gate (Step) Time command is not reserved, which means that no matter how many times the command is repeated, it is not omitted. It is possible that the Extend Gate (Step) Time command will be inserted in intervals in which other events are omitted by the Reserve command.

* It is possible for multiple Extend Gate (Step) Time commands to appear in succession, but because this is only a 1-byte command with no data or Step Time other than the command byte, it becomes meaningless if it is omitted, and therefore it is not subjected to reserve. Furthermore, due to the converter program provisions, it is not possible even to know how many Extend Gate (Step) Time commands there are until actual output to the file. Therefore there is no means of reserving it. (A command already written to a file cannot be reserved.)

83H: Reference

83H (1 byte)	Byte position from top of track for upper byte of top event of referenced event group	Lower byte	Number of events referenced (1 byte)

This command does not appear in a temporary file; it appears only in complete files. The command reduces the amount of data by omitting the second and subsequent events when the same event group (three events or more) appears multiple times in a track.

The two bytes following the command number 83H indicate the position of the reference destination. As explained earlier, the reference destination must be positioned before the command.

The last byte indicates the number of events to be referenced. The maximum number of events is 255. The Extend Gate (Step) Time command is not counted in the number of events. All other events are counted in the number of events.

None of the data is omitted from the top event of the event group specified in the reference destination of this command. That is,

- It is not reserved by the Reserve command
- It is not subject to Step Time omission (all events, not only Note On)
- It is not subject to omission of Musical Interval, Velocity, or Gate time

When the events at the reference destination become the object of the Extend Gate (Step) Time command (described later in this document), the reference destination of the Reference command becomes the position of the first Extend Gate (Step) Time command. Therefore a special data restoration operation is not required to read only the specified number of events from the reference destination.

84H: Loop Start

84H (1 byte)	Step (1 byte)

This command indicates the start position of an endless loop. Because this is not automatically judged and inserted, the user must clearly specify the starting point of the loop. When the user inserts a no. 31 Control Change (not yet defined in the MIDI standard) during editing on a MIDI sequencer, the converter converts it to this command.

If the play program reads 8EH (End of Track) in data that has been converted by the converter and 84H (Loop Start) appears before the end of the track, the program returns to that point and continues play.

If multiple Number 31 Control Changes appear in one track, the converter displays an error message and stops the conversion process. Therefore there can be no more than one of these commands in one track in a converted file. Furthermore, the existence of no more than one of these commands means that it will not be affected by the Reference command or the Reserve command. Therefore the starting point of the loop will not be omitted by the Reference command or the Reserve command.

Loop and Tempo Changes

Loops must be specified independently for each track. In other words, a common loop cannot be specified for all tracks. Therefore the tempo cannot be changed in a loop.

In a standard MIDI file a loop command is not defined, but loops are required when actually composing. When defining a loop command, one common for all tracks and those independent for each track cannot exist together— for example, when the ranges of an overall loop and of track-independent loops overlap. Also, a complex loop may result in a different tune being played during play while composing on a MIDI sequencer and after conversion.

From the standpoint of effectiveness, track-independent loops are considered more versatile, and therefore loops common to all tracks have not been included.

Therefore, because tempo changes are not done independently for each track, the tempo cannot be changed as independent loops for each track are put together.

8EH: End of Track

8EH (1 byte)

This command indicates the end of a track.

91H: Poly-Key Pressure

91H (1 byte)	Interval (1 byte)	Value (1 byte)	Step (1 byte)

Musical Interval and Value are expressed with 7 bits, which is the same as in the MIDI standard.

When the duration (Step Time) until the next event is the same as in the preceding event (including Note On), Step is omitted. In this case, the 7th bit of Musical Interval is set to ON. As a result, the numerical value of Musical interval becomes greater than 80H.

92H: Program Change

92H (1 byte)	Tone Number (1 byte)	Step (1 byte)

Tone Number is expressed with 7 bits, which is the same as in the MIDI standard.

The duration (Step Time) until the next event is omitted when it is the same as in the preceding event (including Note On). In this case, the 7th bit of Tone Number is set to ON. As a result, the numerical value of Tone Number becomes greater than 80H.

93H: Channel Pressure

93H (1 byte)	Value (1 byte)	Step (1 byte)

Value is expressed with 7 bits, which is the same as in the MIDI standard.

The duration (Step Time) until the next event is omitted when it is the same as in the preceding event (including Note On). In this case, the 7th bit of Value is set to ON. As a result, the numerical value of Value becomes greater than 80H.

94H: Pitch Bend (14-Bit Expression)

94H (1 byte)	Value Upper Byte (MS)	Lower Byte (LS)	Step (1 byte)

This command is not output when fade compression of Pitch Bend is specified in the preference settings. Value is expressed with two 7-bit bytes, which is the same as in the MIDI standard (expression in the MIDI standard, however, is replaced with an upper byte and a lower byte).

The duration (Step Time) until the next event is omitted when it is the same as in the preceding event (including Note On). In this case, the 7th bit of the Value upper byte is set to ON. As a result, the numerical value of the Value upper byte becomes greater than 80H.

95H: Pitch Bend (7-Bit Expression)

95H (1 byte)	Value (1 byte)	Step (1 byte)

This command is output only when fade compression of Pitch Bend is specified in the preference settings. D4 of the header flag becomes "1" at that time.

Value is expressed with 7 bits, which is the upper byte of the two 7-bit bytes normally used for Pitch Bend expression in MIDI. Only -64 to +63 can be expressed, but since most of the keyboards available on the market output only about 64 steps up and down anyway, there should be no problem with fade in actual use when only the upper byte is used.

The duration (Step Time) until the next event is omitted when it is the same as in the preceding event (including Note On). In this case, the 7th bit of Value is set to ON. As a result, the numerical value of Value becomes greater than 80H.

A0H: Control Change

A0H (1 byte)	Controller Type (1 byte)	Value (1 byte)	Step Time (1 byte)

Value is expressed with 7 bits, which is the same as in the MIDI standard. Step time (duration until the next event) is omitted when it is the same as in the preceding event (including Note On). In this case, the 7th bit of Controller Type is set to ON. As a result, the numerical value of Controller Type becomes greater than 80H.

There are many types of controllers, but those most commonly used are defined as independent events. This makes it possible to express Control Change in 3 bytes instead of the normal 4 bytes.

The following table lists the Control Changes that are independent events.

Top Byte of Event	Controller
A1H	Modulation
A2H	Breath control
A3H	Foot control
A4H	Main volume
A5H	Panpot
A6H	Expression

A1: Modulation
A2: Breath control
A3: Foot control
A4: Main volume
A5: Panpot
A6: Expression

A1H - A6H (1 byte)	Value (1 byte)	Step Time (1 byte)

Controllers often used from A0H (Control Change) have become independent events.

B0H - B7H: Extend Gate Time

B0H - B7H (1 byte)

Extends the Gate Time of the Note On appearing last. Note On events can only have a 9-bit Gate Time. Therefore, in the case of a Note On event with a Gate Time longer than can be expressed with 9 bits, this command must be used to extend the Gate Time of the next Note On event.

This Extend Gate Time command is not output before events other than Note On. Also, the Extend Gate Time command is not output before Note On events for which Gate Time is omitted, regardless of the length of the Gate Time of the Note On.

The amount of extension is provided in stages, each of which is an independent command, as shown in the following table.

Command Number	Length of Extension (Hex)	(Decimal)
B0H	200H	512
B1H	400H	1024
B2H	600H	1536
B3H	800H	2048
B4H	A00H	2560
B5H	C00H	3072
B6H	E00H	3584
B7H	1000H	4096

B8H - BFH: Extend Step Time

B8H - BFH (1 byte)

Extends the Step Time of the immediately following event. The expression width of Step Time is 10 bits for Note On events and 8 bits for other events. The amounts of Step Time expressed by these widths are 3FFH and FFH, respectively. In the case of a Step Time that must be longer than this, the Extend Step Time command is used to extend the Step Time of the next event.

Because Note On events can be expressed with Step Times up to 3FFH even without extension, an extension of less than 400H is not possible before Note On events.

The Extend Step Time command is not output before events for which Step Time is omitted, regardless of the length of the Step Time of the event

The amount of extension is provided in stages, each of which is an independent command, as shown in the following table.

Command Number	Length of Extension (Hex)	(Decimal)
B8H	100H	256
B9H	200H	512
BAH	400H	1024
BBH	600H	1536
BCH	800H	2048
BDH	1000H	4096
BEH	1800H	6144
BFH	2000H	8192

5.0 Abbreviation Rules

The top byte (command number) of events and the Step Time are not omitted in the following cases.

- Top event of event groups referenced by the Reference command
- Next event following the Reference command
- Next event following the Loop Start command
- Top of each track
- Next event following the Rest command

These cases may occur simultaneously. It is also possible that the ranges of event groups referenced by the Reference command may overlap.

In cases other than those above, the Step Time may be omitted when it is the same as the preceding event. In this case, the 7th bit of the data byte following the top byte (command number) of the event is set to ON

If the preceding event and event number are the same in cases other than those above, and if the event is other than Note On, the top byte (command number) of the event may be omitted by the Reserve command.

SEGA

SEGA OF AMERICA, INC.
Consumer Products Division

Tone Editor
User's Manual

Doc. # ST-68-R1-042594

NOTICE

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.

2.

3. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

4. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

5. No license is granted by implication or otherwise under any patents or other rights of any third party or SEGA Enterprises, Ltd.

6. This document is confidential. By accepting this document you acknowledge that you are bounded by the terms set forth in the non-disclosure and confidentiality agreement signed separately and/in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

(4/20/94 - 001)

Table of Contents

(This page was blank in the original Japanese document)

1.0 Summary

This program permits the display and editing of tone parameters using an SCSP connected to the SCSI port of an Apple Macintosh computer. This program handles the following seven operations.

- **File Management**
  Manages file tasks including file input/output and saving.

- **Editing**
  Performs data editing operations including undo, cut and paste, and insert.

- **Number Processing**
  Increases voice, layer, mixer, velocity, PEG, and PLFO values.

- **Window Operation**
  Opens/closes Mixer, Velocity, PEG, PLFO, FM, and MONITOR windows.

- **FM Processing**
  Updates all layers that have the same start and end notes as the selected layer.

- **Preferences**
  Displays the SCSI ID, downloads the SCSPBIN format, and displays voice data.

- **SCSI Operations**
  Handles SCSI operations on the Macintosh during I/O of tone data to the Macintosh from the SCSP through the SCSI port.

2.0 Terminology

- **SCSP Format**
  The file containing all data required for display on the Macintosh.

- **SCSPBIN Format**
  The file containing the minimum data required to play a sound on the SCSP. The data in this file is actually downloaded to the SCSP.

- **Voice**
  The tone. The number associated with this window is the same as the MIDI program change number.

- **Layer**
  The sound data. Multiple layers are combined to form one voice. With FM, each module and each carrier is treated as a layer.

- **Wave**
  One type of sound data; used in reference to the PCM data of an actual sound.

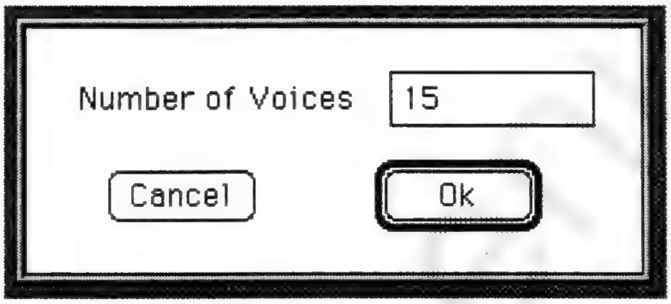
3.0 Tutorial

This tutorial describes the minimum operating procedure required to play a sound.

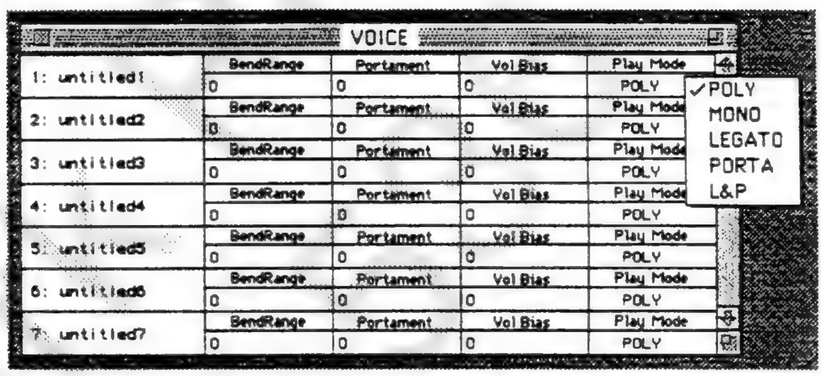
Voice Window Display

1. Start the tone editor.
   The tone editor first checks for the presence of the SCSP board in one of the Macintosh slots; if the board is not found, the program quits.

2. Select New from the File menu.
   A dialog box requesting the number of voices will be displayed.



3. Enter desired number of voices. In this example, enter "1". The Voice window will be displayed.



   Note that only one line is displayed at this time because the specified number of voices was "1". Also note that the number of voices can be changed later.

Operation Example

- **Voice Window Setup Items**

  The Voice window setup items are:
  - Voice name
  - Bend range
  - Portamento time
  - Volume bias
  - Play mode

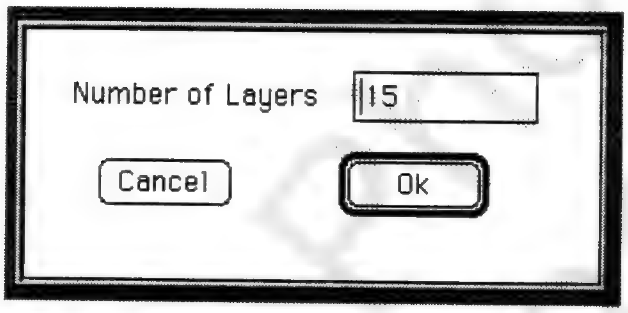
  Except for play mode, these item settings can be changed by clicking the number. For play mode, a pop-up menu is displayed.

  1. **Enter the Number of Voices**

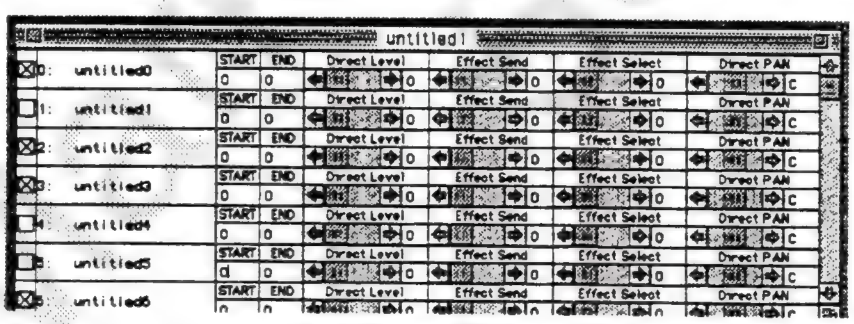
     Double-click the voice name to display a dialog box enabling the voice name to be changed. The default filename for the first new file created is "untitled0". Double-click the name to display the dialog box, and enter any desired name. Managing the voices later may be simplified by assigning a name appropriate to the tone you create. Click the **OK** button to accept the name change and close the dialog box. The new name just entered should be displayed in the voice name area. The next step is to create the tone.

  2. **Set the Number of Layers**

     Double-click the number before the voice name to display the dialog box for setting the number of layers.

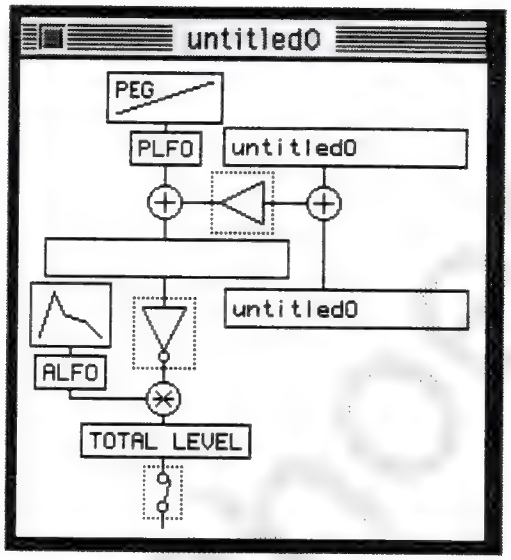


Enter the number of layers as done for the number of voices. Enter "1" now. The Layer window will be displayed.

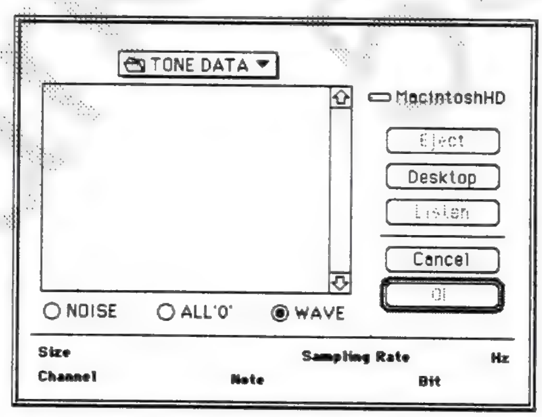


Again, only one line will be displayed because there is only one layer; the number of layers can be changed later.

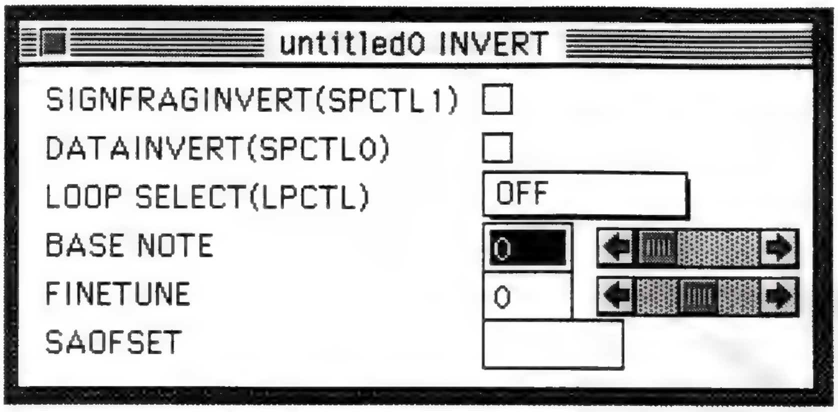
3. **Operate the Layer Window**
The Layer window functions as the mixer, making it possible to adjust the output and set the channel and level input to the effector. To determine the minimum operation required to output a sound, set the output to maximum and decide the range of the note of the MIDI device playing the sound. Click the END number. The insertion mark will be displayed. Now, enter "127". Next, move the DirectLevel slide bar all the way to the right. The number beside the slide bar should be 7. Enter a name for the layer. The procedure is the same as entering the voice name. This completes the output settings. All that remains is to set the detailed parameters. To do this, double-click the number beside the layer name. The Slot window will be displayed.



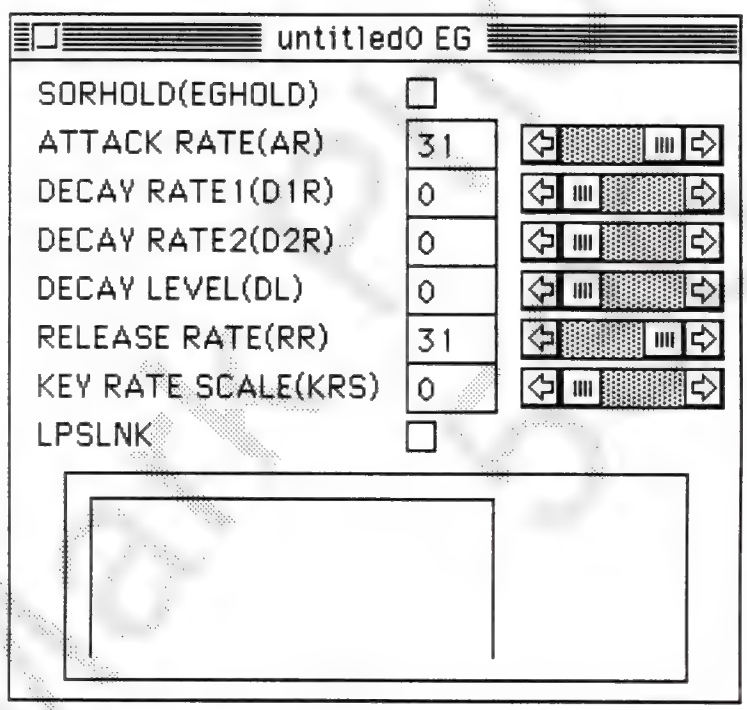
4. **Operate the Slot Window**
The parameters are divided into individual blocks, each with its own special window. First, only the minimum parameters required to play a sound will be set at this time. To display the wave data (AIFF data) input window, double-click the blank rectangle in the middle.



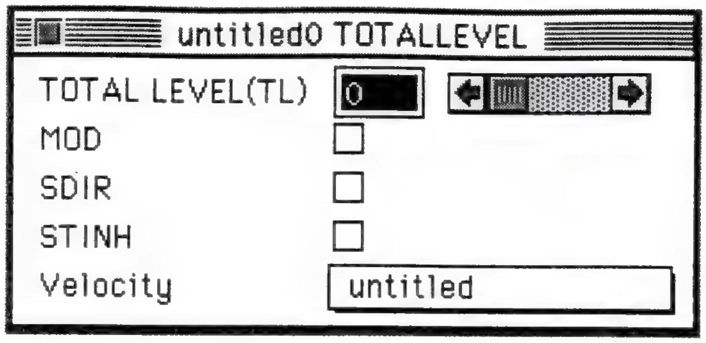
Select the AIFF file and click **OK**. With the Slot window still displayed, the AIFF filename will appear in the previously empty rectangle. If there is a loop in the selected wave data, double-click the Slot window inverter (small triangle with a circle around it). The Inverter window will be displayed.



Select the loop mode from the LOOP SELECT pop-up menu. Next, double-click the picture of a mountain in the Slot window. The EG window will be displayed. Move the AR and RR slide bars all the way to the right ("31" will be displayed).

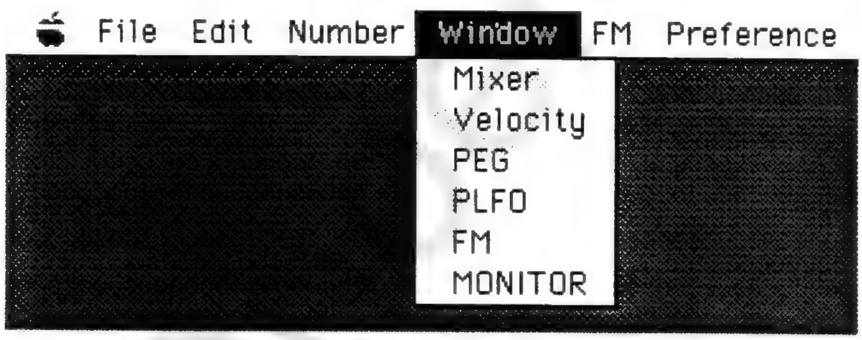


For the final setting, double-click the TOTAL LEVEL in the Slot window. The TOTAL LEVEL window will be displayed.

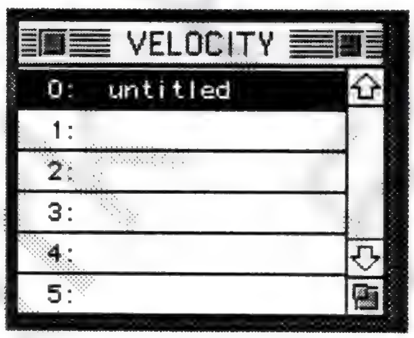


The velocity is selected from a pop-up menu, but is still not set.

5. **Set the Velocity**
To set the velocity, first close the TOTAL LEVEL window. Select Velocity from the Window menu.

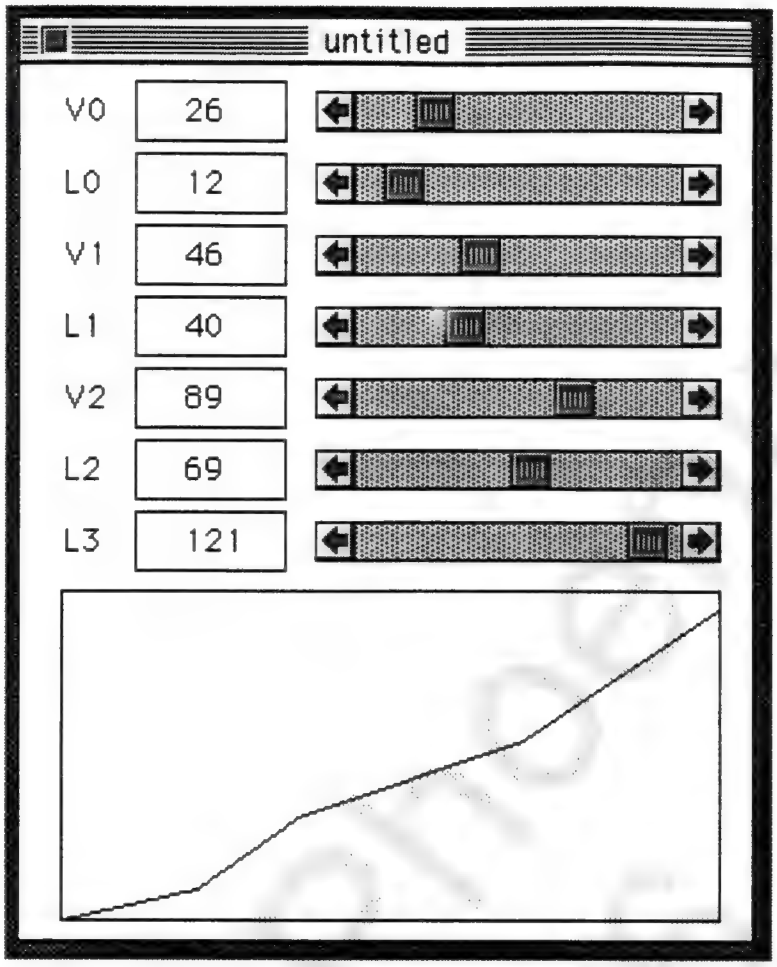


The Velocity window will be displayed.



Double-click "untitled" to enter a name, and double-click the number beside the name to display the Velocity Edit window.

Double-click number 0 to display the following.



Move the L0, L1, L2, and L3 slide bars all the way to the right. Close the Velocity Edit and Velocity windows to unclutter the screen. Now display the TOTAL LEVEL window again. The Velocity pop-up menu should now contain the name just assigned. If more than one velocity name was entered using the Velocity window, the other names can be selected from the pop-up menu.

6. **Play a Sound**

A sound can now be played. Try playing the MIDI device connected to the SCSP board with program change 0.

4.0 File Descriptions

The following files are used with the Tone Editor.

- **SCSP Data Files**
  Macintosh files containing detailed information. See the Appendix for details.

- **SCSPBIN Data Files**
  Files containing only the data that is actually loaded into 68000 memory and the header data required to create Macintosh files. See the Appendix for details.

- **Wave Edit Data Files**
  AIFF format files created with the Save operation.

- **Alchemy Files**
  AIFF format files created by Alchemy.

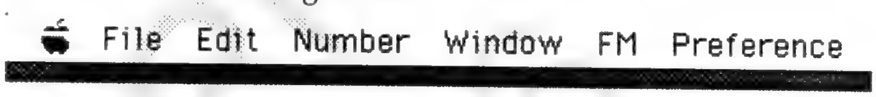
- **Sound Designer Files**
  AIFF format files created by Sound Designer.

5.0 Menu Summary

This chapter introduces the menu bar, pull-down menus, and Master Volume window of the Tone Editor.

Menu Bar

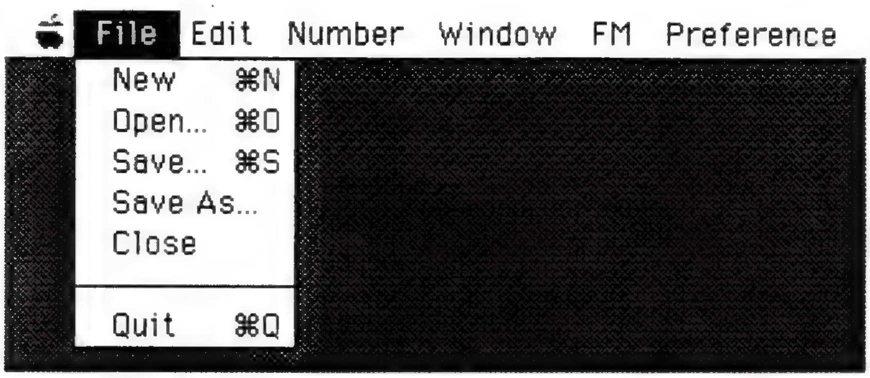
The menu bar contains the following menus.



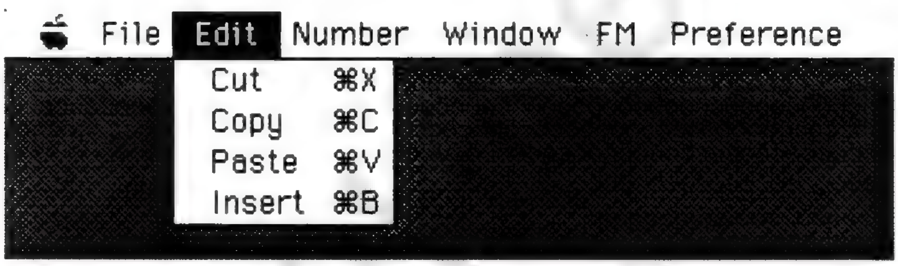
Pull-down Menus

- **Apple Menu**
  This is the standard Apple menu.

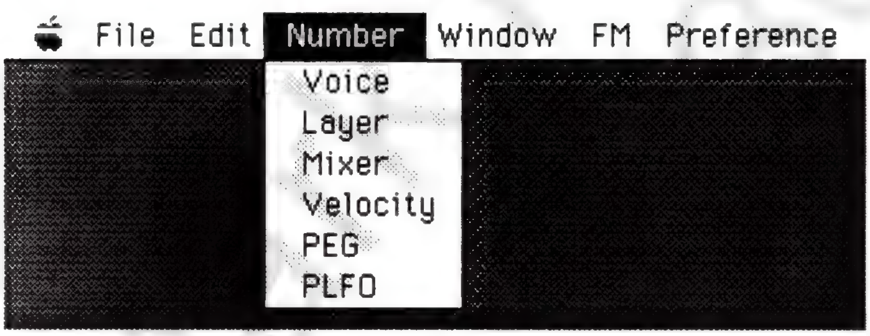
- **File Menu**

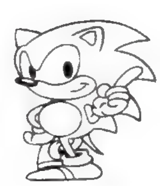


- **Edit Menu**

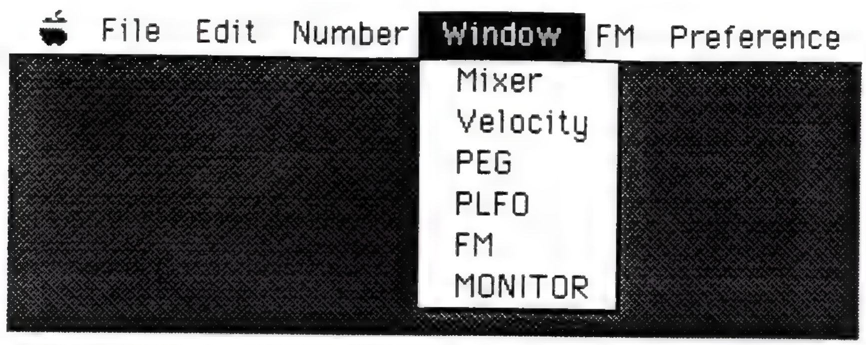


- **Number Menu**



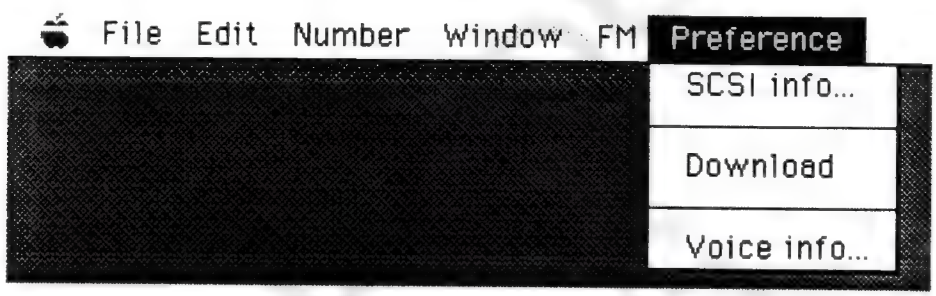


- **Window Menu**



- **FM Menu**

- **Preference Menu**



Master Volume Window

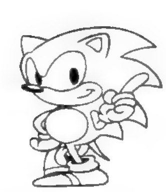
The Master Volume window is always displayed. The master volume can be changed while the master volume setting is displayed.

6.0 Function Details

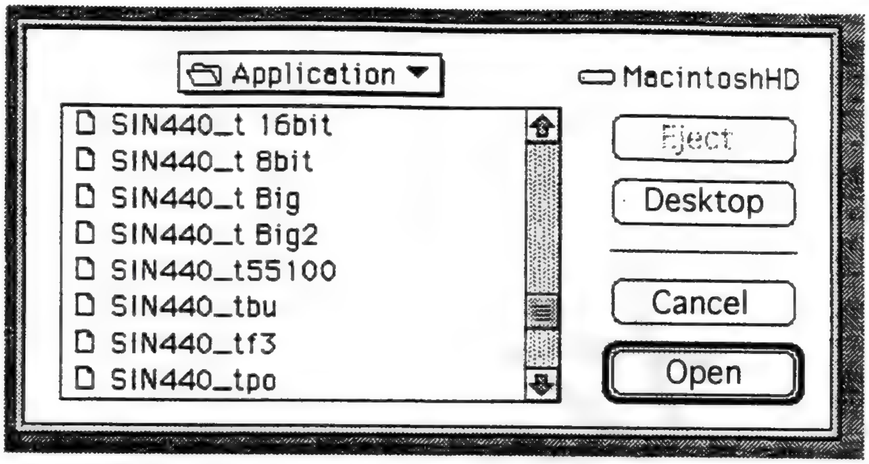
This chapter explains the function of each menu item.

File Menu

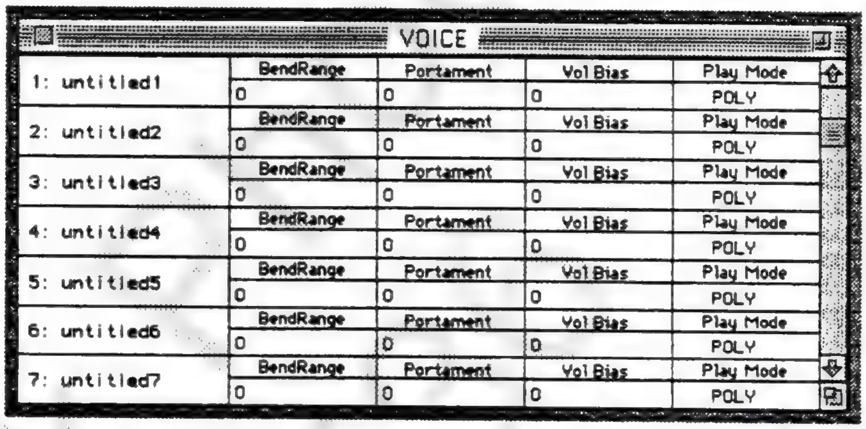
- **New**
  Opens a new tone library. To open a new tone library, open the Number of Voices window, set the number of voices, and click **OK**.

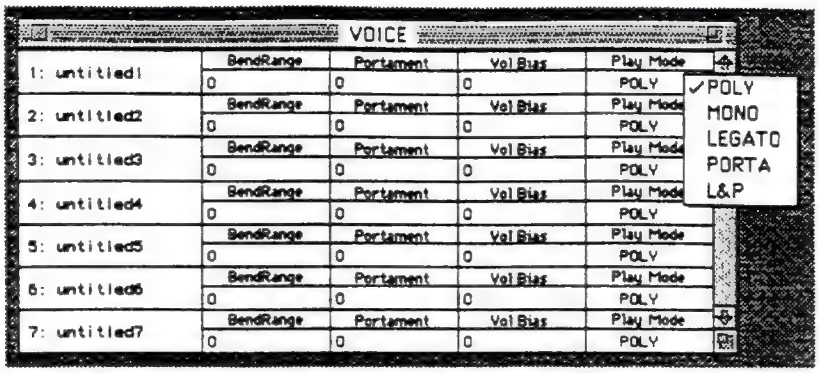


- **Open**
  Opens an existing tone format file. Selecting "Open" displays the following screen:

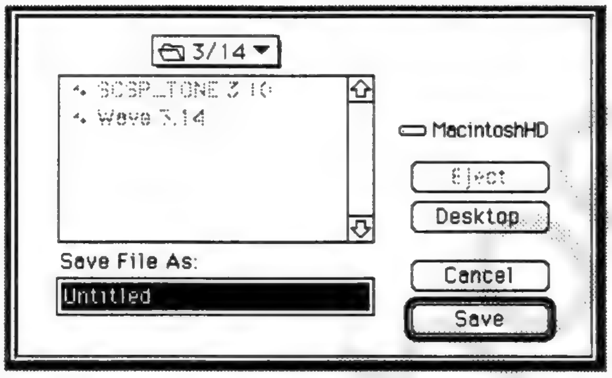


The screen lists only the SCSP format files. Only the files displayed in the screen can be loaded. Double-click a filename to open the Voice window (see page 26).

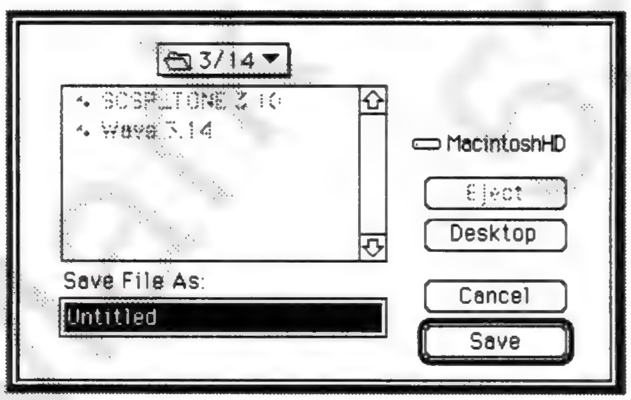




- **Save**
  Saves the SCSP file currently in use.

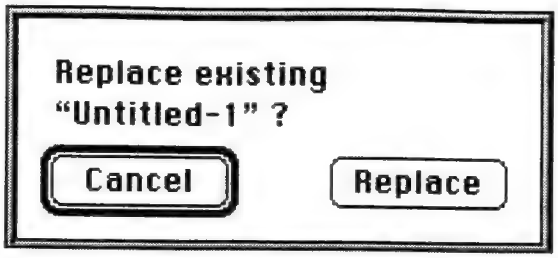


- **Save As**
  Saves the SCSP file currently in use under a different filename (this allows the user to change the filename). The following screen opens.

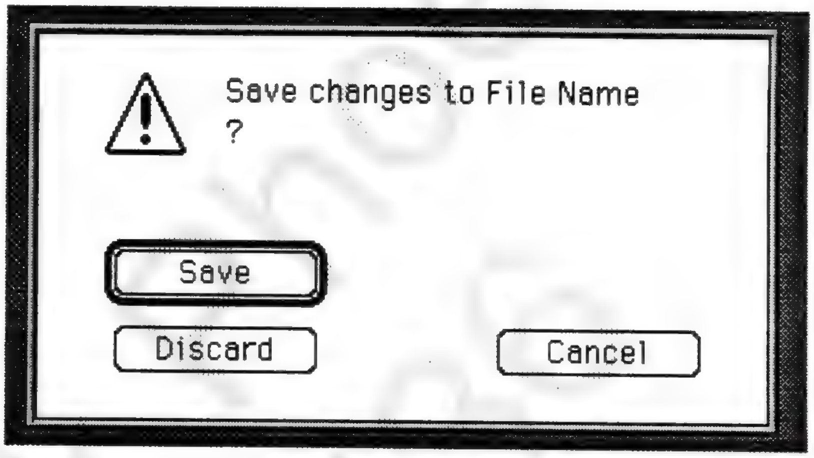


Below is one of the buttons used in this window.

- Save
Displays the Replace dialog, and saves the file if the Replace button is pressed.



- **Close**
  Closes the file currently in use. An alert is displayed if the user attempts to close the file without saving it after changes have been made.
  Windows beneath the active window are also closed automatically.
  When the Slot window is closed, all Parameter windows are also closed.
  When the Layer window is closed, the Slot window is also closed.
  When the Voice window is closed, an alert is displayed if changes have been made.



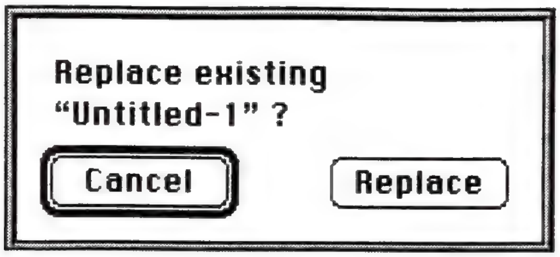
The following buttons are used in this window.

- Save
Saves the file.

- Discard
Closes the window without saving the changes.

- Cancel
Returns to the Edit window.

- **Quit**
  Terminates the Tone Editor and returns to the Finder. If any windows are open, an alert is displayed.



Edit Menu

- **Cut**
  When in the Layer window, cuts the layer from the format to the clipboard. When in the Voice window, cuts the layer associated with the voice chunk, and saves the voice chunk data and the associated layer chunk data to the clipboard. When in the PEG, Velocity, or PLFO windows, cuts the data type of the window from the format to the clipboard.

- **Copy**
  When in the Layer window, copies the layer chunk data to the clipboard. When in the Voice window, copies the voice chunk data and the associated layer chunk data to the clipboard. When in the PEG, Velocity, or PLFO window, copies the data type of the window to the clipboard.

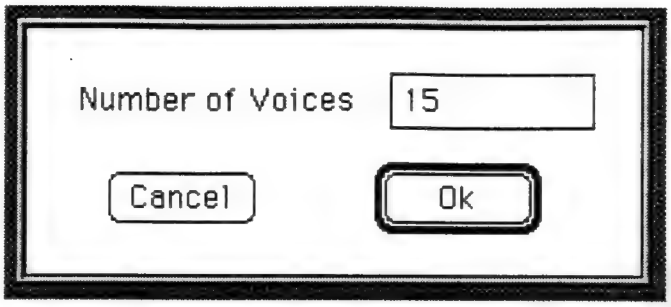
- **Paste**
  Pastes the data from the clipboard. The clipboard data must match the data type of the window being pasted into. Paste operations are overwrite only.

- **Insert**
  Adds one more voice, layer, or mixer.

Number Menu

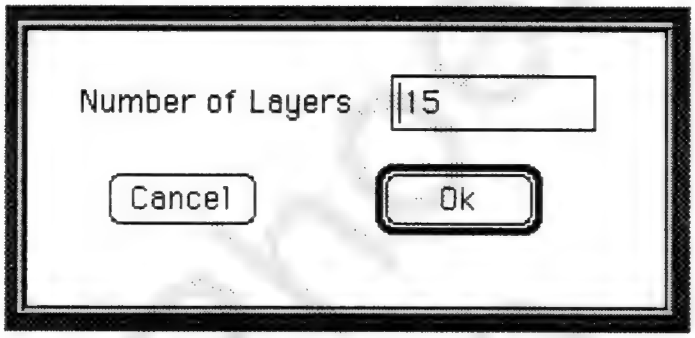
Number menu options are used to increase the corresponding value.

- **Voice**
  Sets the number of voices. The following dialog box opens.



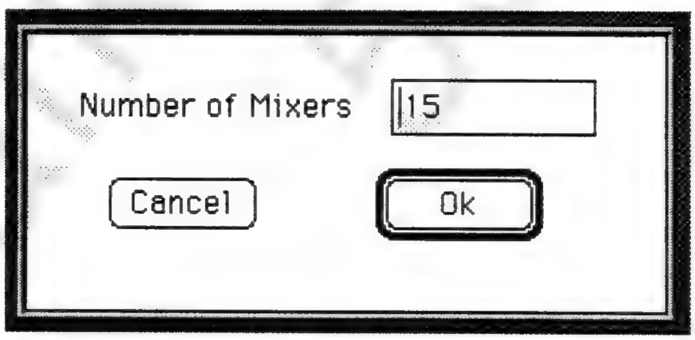
It is not possible to set a value less than the current number of voices.

- **Layer**
  Sets the number of layers. The following dialog box opens.



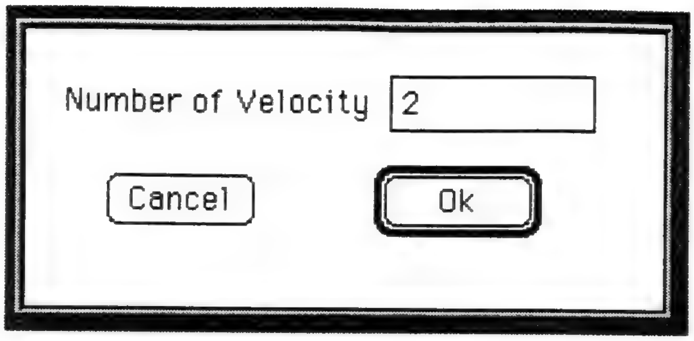
It is not possible to set a value less than the current number of layers.

- **Mixer**
  Sets the number of mixers. The following dialog box opens.



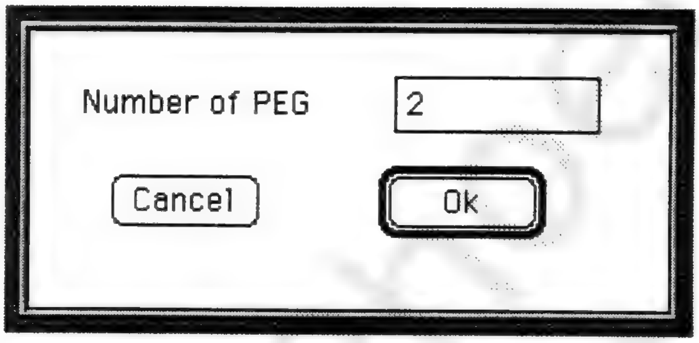
It is not possible to set a value less than the current number of mixers.

- **Velocity**
  Sets the velocity number. The following dialog box opens.



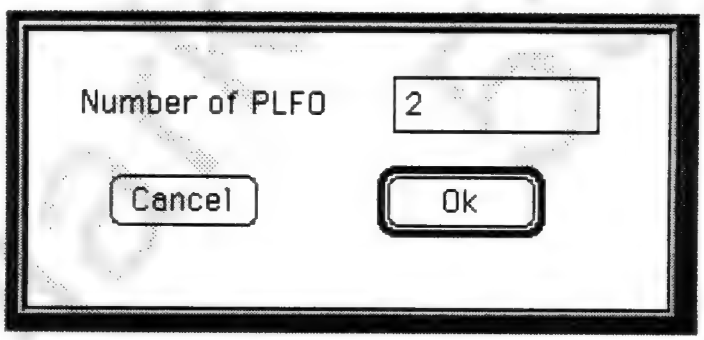
It is not possible to set a value less than the current velocity setting.

- **PEG**
  Sets the PEG number. The following dialog box opens.



It is not possible to set a value less than the current PEG setting.

- **PLFO**
  Sets the PLFO number. The following dialog box opens.

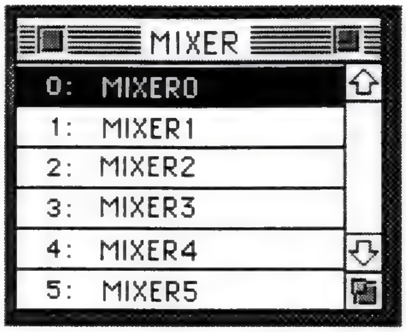


It is not possible to set a value less than the current PLFO setting.

Window Menu

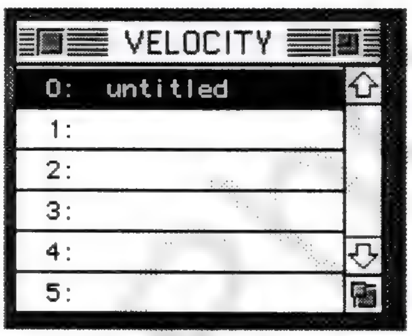
Each Window menu item opens the corresponding closed window or closes the corresponding open window.

- **Mixer**
  Opens/closes the Mixer window.



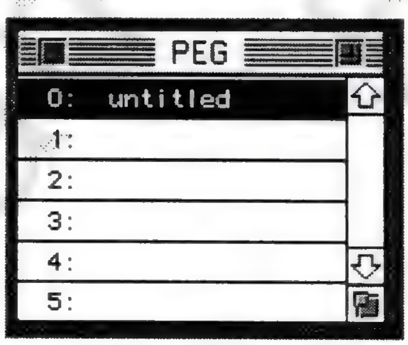
For information on the Mixer window, see page 38.

- **Velocity**
  Opens/closes the Velocity window.



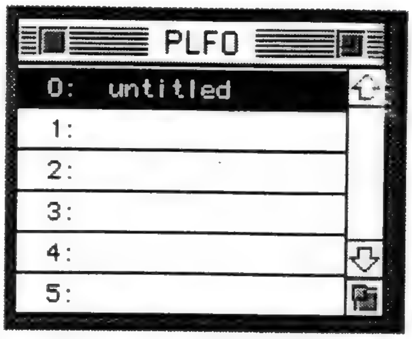
For information on the Velocity window, see page 39.

- **PEG**
  Opens/closes the PEG window.



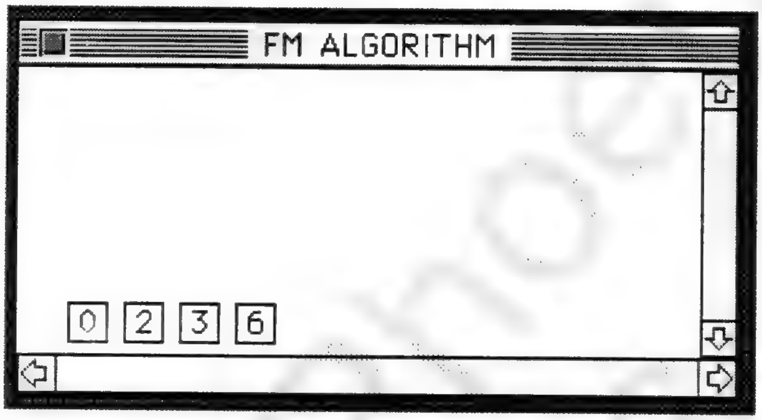
For information on the PEG window, see page 41.

- **PLFO**
  Opens/closes the PLFO window.



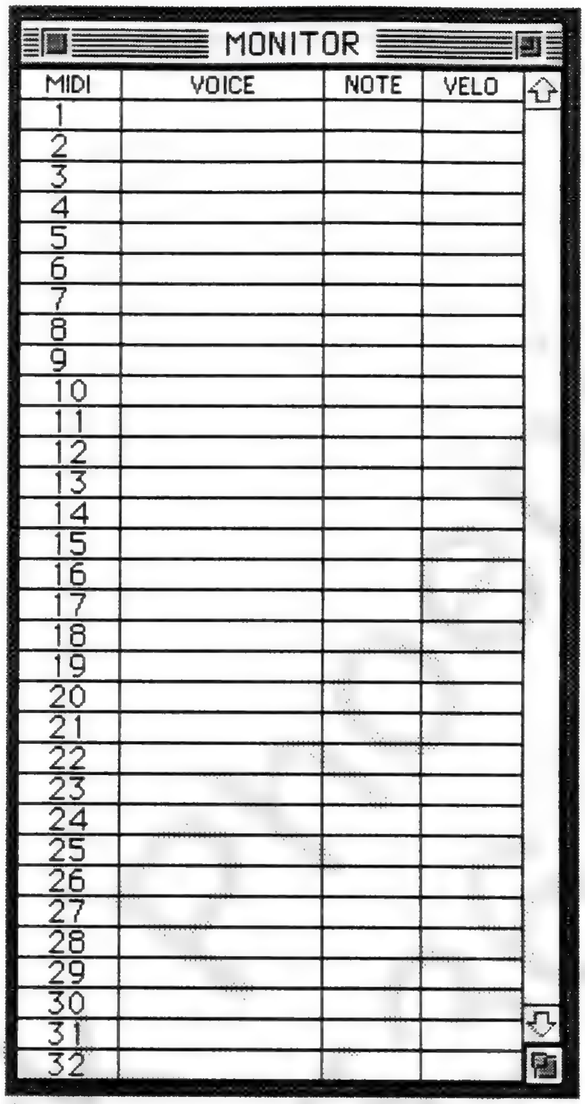
For information on the PLFO window, see page 43.

- **FM**
  Opens/closes the FM Connection window.



For information on the FM Connection window, see page 45.

- **MONITOR**
  Opens/closes the MONITOR window.



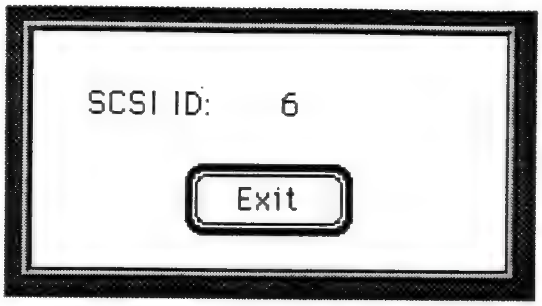
For information on the MONITOR window, see page 46.

FM Menu

- **UPDATE**
  Updates layers that have the same start and end notes as the layer selected in the Layer window.

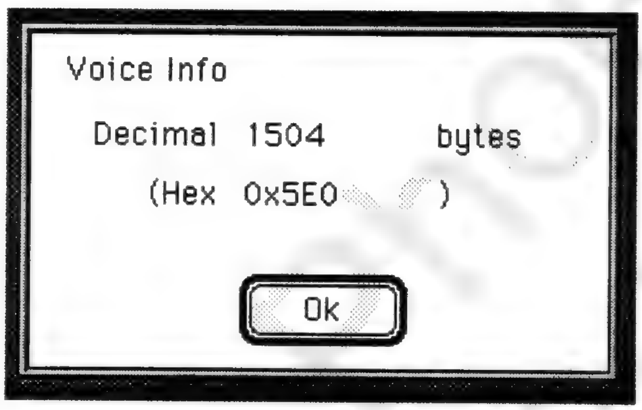
Preference Menu

- **SCSI Address**
  Displays the current SCSI ID.



- **Download**
  Downloads the SCSPBIN format for tone data to the SCSP board.

- **Voice Info**
  Displays the total number of bytes in the tone data being edited.

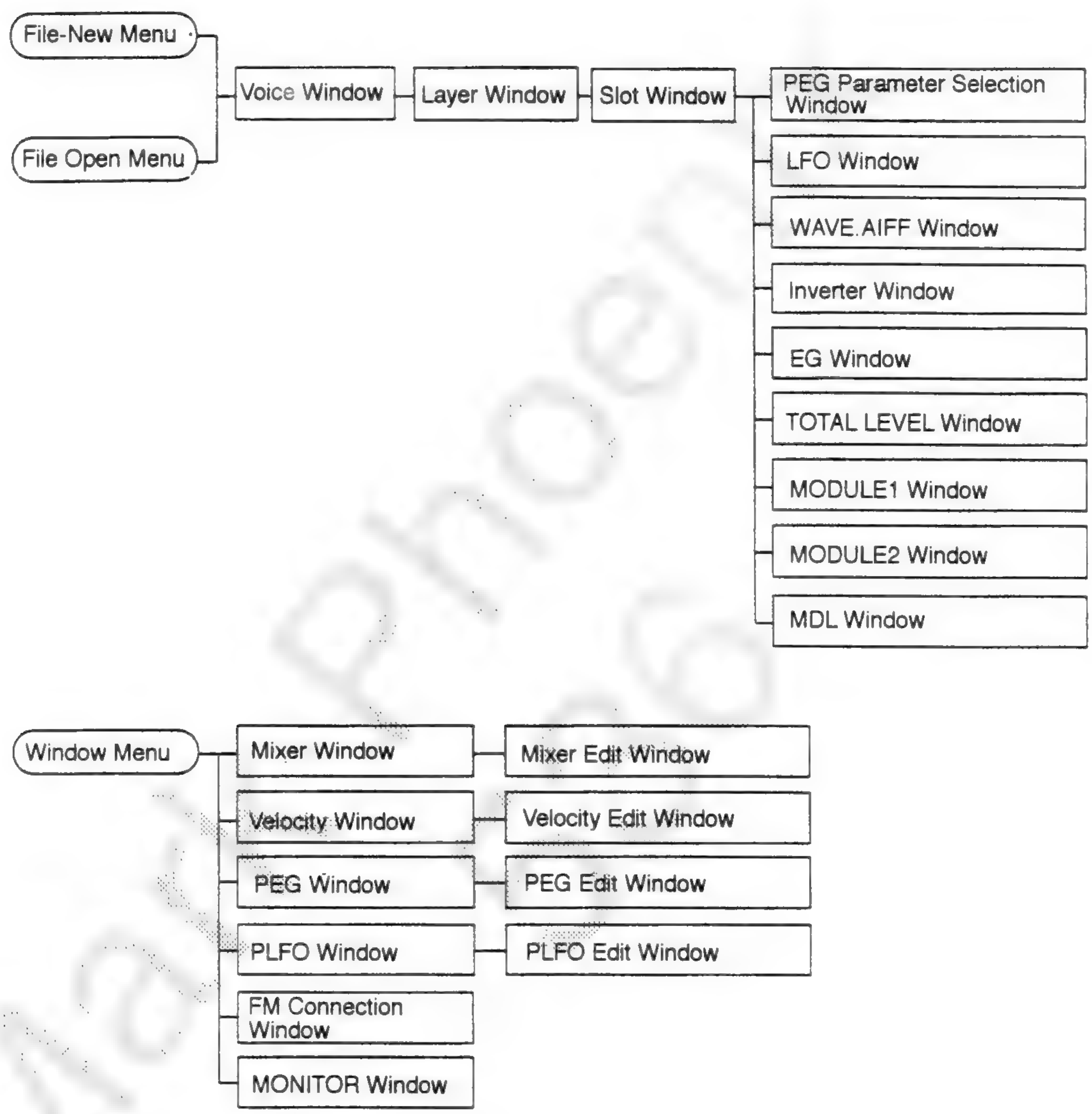


7.0 Windows

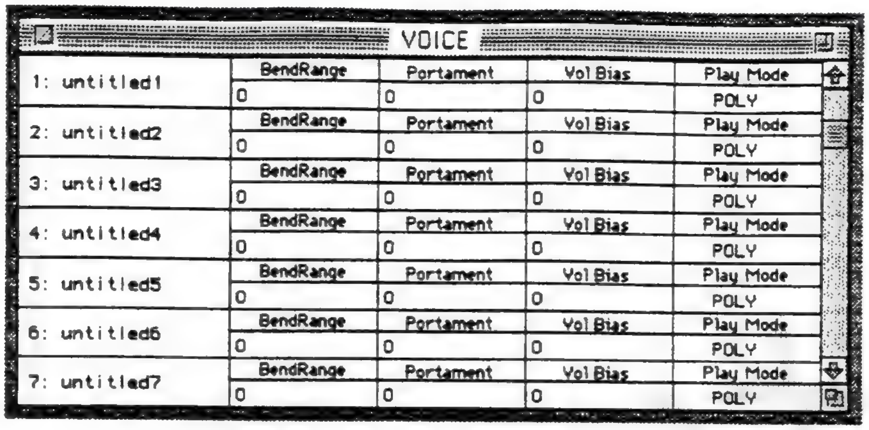
This chapter explains the parameters that can be set for the functions of each window.

Window Relationships

The window relationships are shown below.



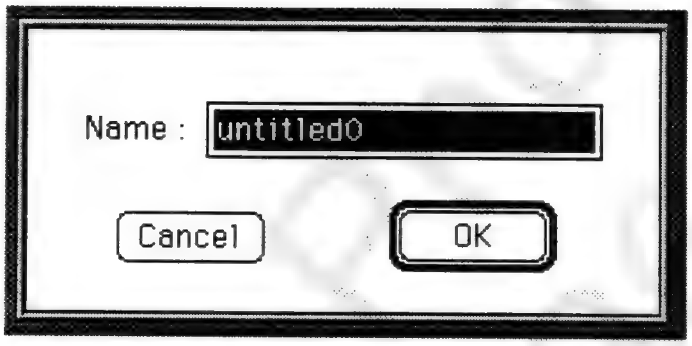
Voice Window



VOICE				
1: untitled1	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY
2: untitled2	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY
3: untitled3	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY
4: untitled4	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY
5: untitled5	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY
6: untitled6	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY
7: untitled7	BendRange	Portament	Vol Bias	Play Mode
	0	0	0	POLY

The Voice window changes the voice monitor and parameters.
The following parameters can be set from this window.

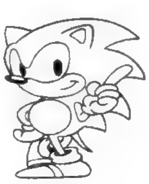
Tone Name
Displays the tone name. Double-click the name to change it; a dialog box will be displayed.



Double-click the number beside the name to open the Layer window (see page 27).

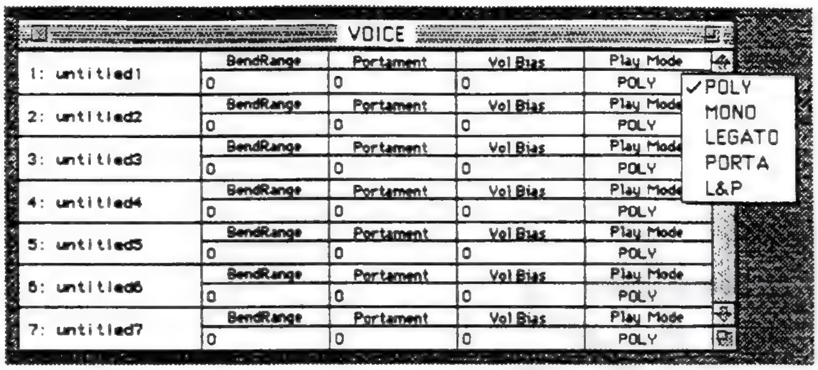
BendRange
Sets the adjustable range of the BendRange value. Values from 0 to 13 can be set. This value expresses the change in pitch as a MIDI note when the pitch Bend wheel is moved to the maximum setting.

Portamento
Sets the portamento time. Values ranging from 0 to 127 are permitted.



VOLBias
Sets the volume bias. Values ranging from -128 to 127 are permitted.

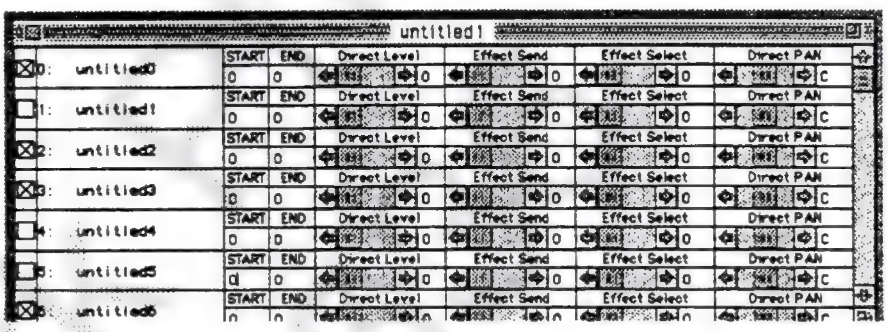
MODE
Displays the play mode.



POLY, MONO, LEGATO, PORTA, L&P mode selections can be made from the pop-up menu. PORTA is an abbreviation for PORTAMENTO. L&P is an abbreviation for LEGATO and PORTAMENTO.

Layer Window

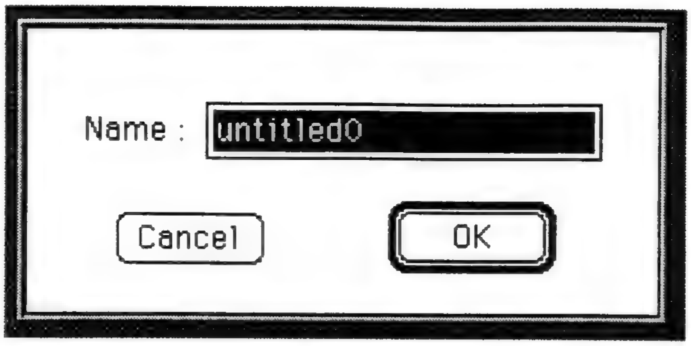
The Layer window is opened by double-clicking the voice name. Layer parameters can be changed from this window.



The following parameters can be set from this window.

FM Connection Carrier Check Box
Base slot used when ON (selected) (if plural boards are installed). This is used in the FM Connection window. For further details, see the FM Connection window section.

Layer Name
Displays the tone name. Double-click the name to change it; a dialog box will be displayed.



Double-click the number to the left of the name to open the Slot window (see page 29).

START
Sets the start note. Values ranging from 0 to 127 are permitted. Also sets which MIDI note is played in this layer. When other MIDI notes come between the START and END notes, the sound from this layer is played.

END
Sets the end note. Values ranging from 0 to 127 are permitted. Also sets which MIDI note is played in this layer. When other MIDI notes come between the START and END notes, the sound from this layer is played.

Direct Level
Sets a direct send. Values ranging from 0 to 7 are permitted; 7 sets the maximum output level.

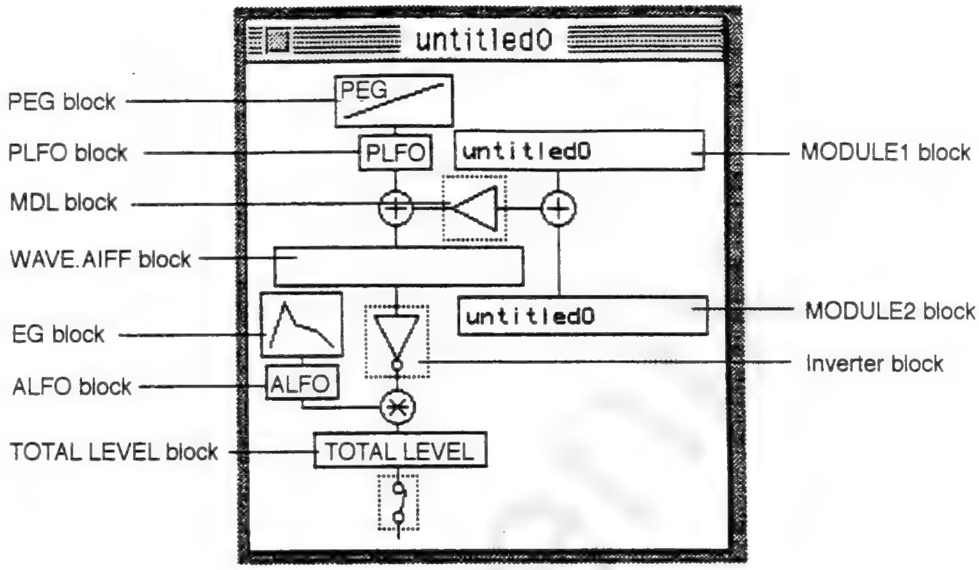
Effect Send
Sets an effect send. Values ranging from 0 to 7 are permitted; 7 sets the maximum output level.

Effect Select
Sets the effect input channel. Values ranging from 0 to 15 are permitted.

Direct PAN
Sets PAN. Values ranging from R16 to C to L16 are permitted.

Slot Window

This window is opened by double-clicking the number in front of the layer name of the Layer window.



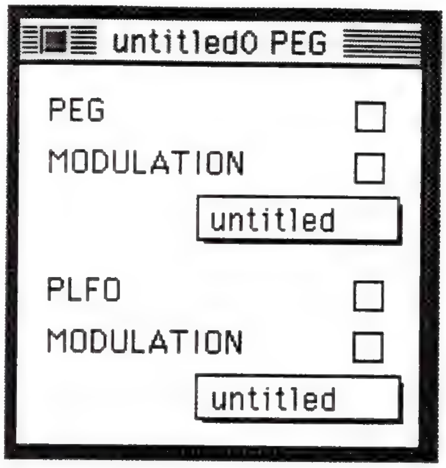
Multiple Slot windows can be displayed simultaneously. The Parameter window applies to all Slot windows, and only one Parameter window can be displayed. SCSP register parameters can be edited, and the Parameter window can be opened.

Double-click the corresponding icons in the Slot window to open the parameter (PEG), PLFO, ALFO, WAVE.AIFF, inverter, EG, SLOTOUT, MODULE1, MODULE2, and MDL blocks. The switches toggle on/off with each click. WAVE.AIFF displays the AIFF filename used in that slot. MODULE 1, 2 display the layer name, and are disabled when the LEVEL (MDL) is 0 (see page 37).

If the Slot window is moved while holding the shift key depressed, the Parameter window opened from that Slot window also moves.

PEG Parameter Selection Window

This window is opened by double-clicking the PEG block in the Slot window.



The Parameter window allows selection of the parameters set from the PEG and PLFO windows using a pop-up menu.

PEG (PEON)
Sets PEG on/off.

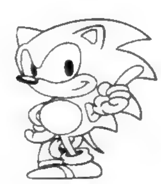
MODULATION (MWE)
Turns PEG adjustment using the MODULATION wheel on/off.

PEG Pop-up Menu
Enables selection of the parameters created in the PEG window.

PLFO (PLON)
Sets PLFO on/off.

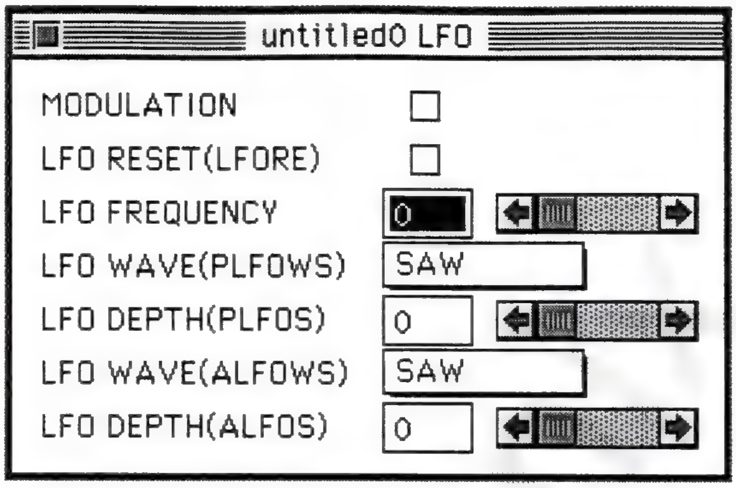
MODULATION (MWL)
Turns PLFO adjustment using the MODULATION wheel on/off.

PLFO Pop-up Menu
Enables selection of the parameters created in the PLFO window.



LFO Window

This window is opened by double-clicking the PLFO or ALFO block in the Slot window.



Parameters relating to hardware LFO can be edited from the LFO window. The following parameters can be set from this window.

MODULATION
Turns the hardware LFO bit on/off.

LFO RESET (LFORE)
Resets LFO each time a note is turned on.

LFO FREQUENCY (LFOF)
Sets the LFO oscillation frequency. Values ranging from 0 to 31 are permitted.

LFO WAVE (PLFOWS)
Sets the shape of the PLFO wave. Available selections are SAW, SQUARE, TRIANGLE, and NOISE.

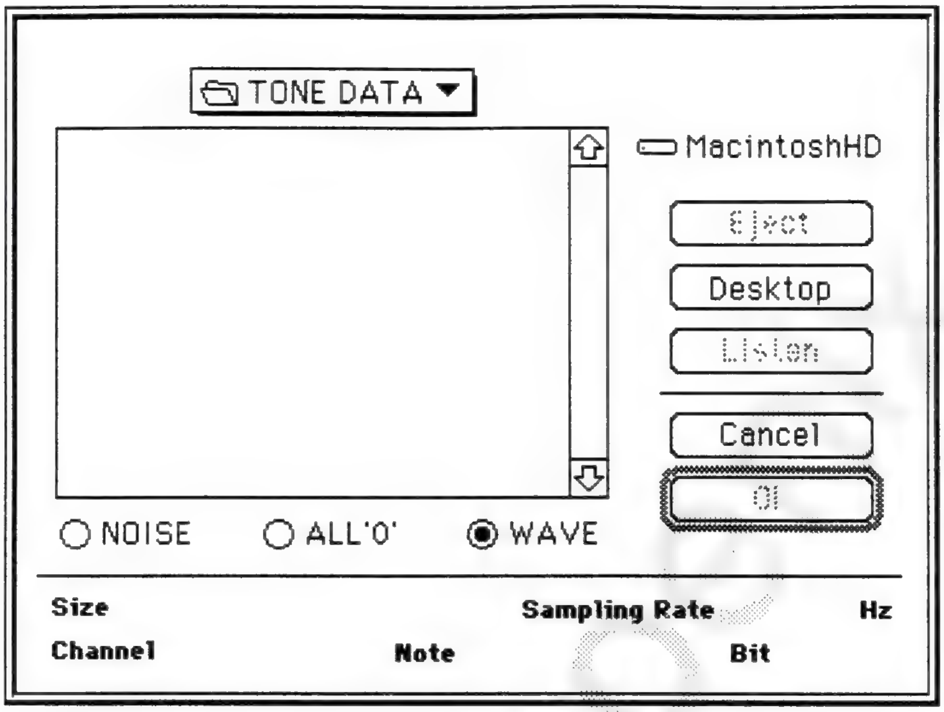
LFO DEPTH (PLFOS)
Sets the effect on LFO pitch. Values ranging from 0 to 7 are permitted.

LFO WAVE (ALFOWS)
Sets the shape of the ALFO wave. Available selections are SAW, SQUARE, TRIANGLE, and NOISE.

LFO DEPTH (ALFOS)
Sets the effect of mixing on LFO EG. Values ranging from 0 to 7 are permitted.

WAVE.AIFF Window

This window is opened by double clicking the WAVE.AIFF block (normally a blank rectangle) in the Slot window.



This window displays the details of the wave to be selected. When the OK button in this window is clicked, the loop start address (LSA) of the selected AIFF file, the loop end address (LEA) of the sound data, and the parameters set with the radio buttons are written in the SCSP format. The following parameters can be set from this window.

NOIS
Uses internally generated data (noise) as the sound input data.

ALL "0"
Uses internally generated data (ALL "0") as the sound input data.

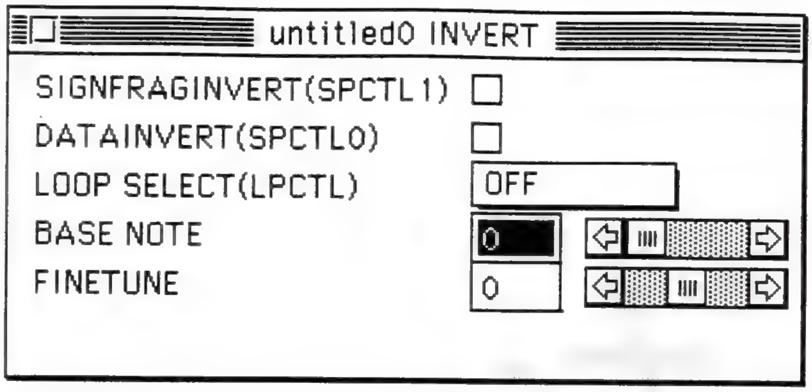
WAVE
Displays the Layer dialog when OK is pressed.

Listen
Plays the selected file through the Macintosh speaker.

OK
Reads the wave.

Inverter Window

This window is opened by double-clicking the inverter block in the Slot window.



This window enables bit inversion of the sound input data, and editing of the loop type, layer sample note, and FINETUNE parameters. The following parameters can be set from this window.

SIGNFRAGINVERT (SPCTL1)
Inverts the sign bit of the sound input data.

DATAINVERT (SPCTL0)
Inverts all bits other than the sign bit of the sound input data.

LOOP SELECT
Selects the loop type from the following options:

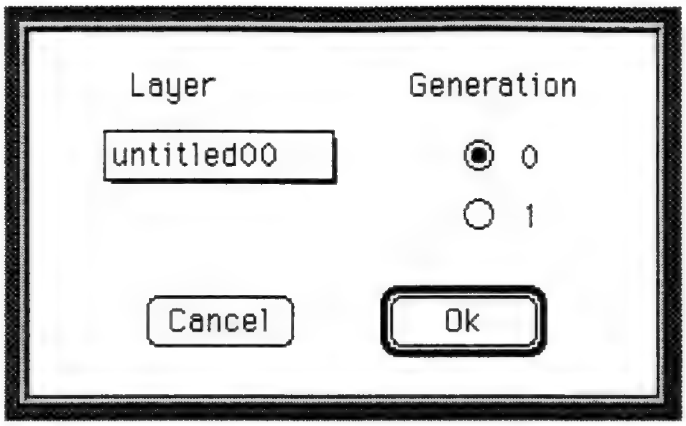
- FORWARD
- REVERSE
- ALTERNATE
- OFF

BASE NOTE
Sets the base note. Values ranging from 0 to 127 are permitted.
When an AIFF file is loaded, the BASE NOTE value in the AIFF file is automatically loaded to set this parameter.

FINETUNE
Sets the fine tune parameter. Values ranging from -64 to 63 are permitted.
When an AIFF file is loaded, the FINETUNE value in the AIFF file is automatically loaded to set this parameter.

MODULE1 Window

This window is opened by double-clicking the MODULE1 block in the Slot window.



This window allows the source to be set as modulation input X. Modulation X of the selected layer is set to the selected generation. The following parameters can be set from this window.

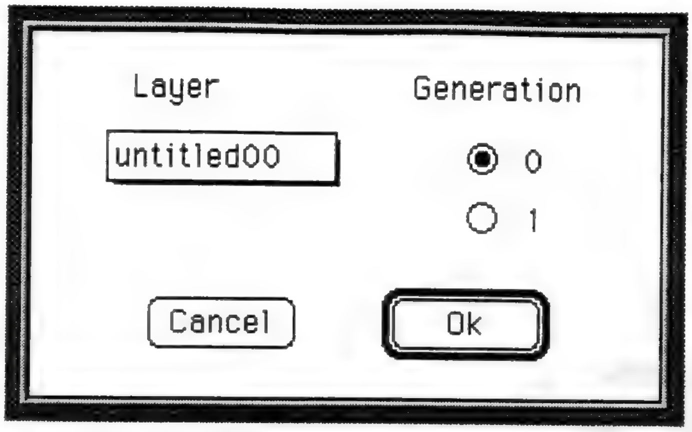
LAYER Pop-up Menu
Displays the layer name.

Generation

- 0
  Latest data
- 1
  Data from the last generation

MODULE 2 Window

This window is opened by double-clicking the MODULE2 block in the Slot window.



This window allows the source to be set as modulation input Y. Modulation Y of the selected layer is set to the selected generation. The following parameters can be set from this window.

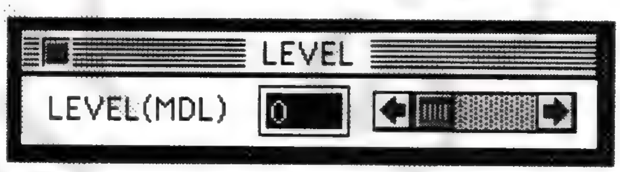
LAYER Pop-up Menu
Displays the layer name.

Generation

- 0
  Latest data
- 1
  Data from the last generation

MDL Window

This window is opened by double-clicking the MDL block in the Slot window.

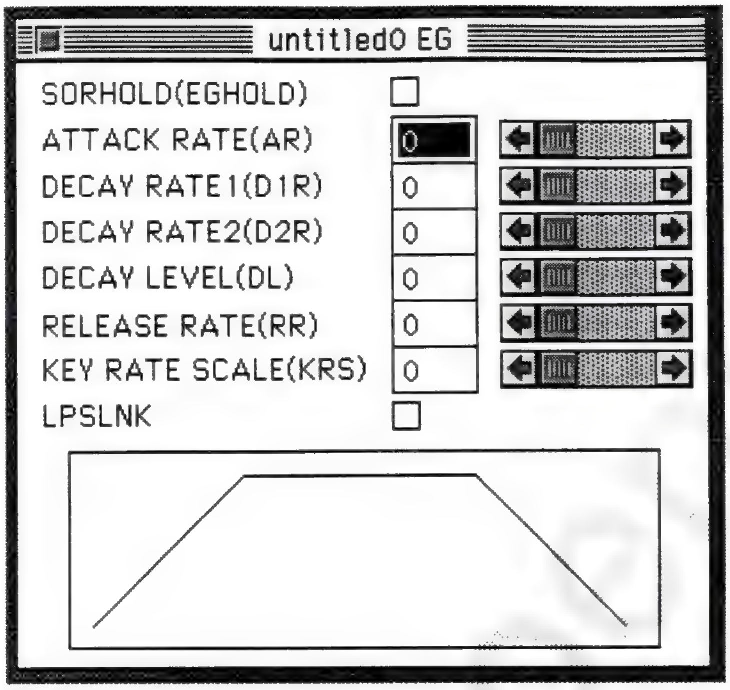


The following parameter can be set from this window.

LEVEL (MDL)
Sets the effect (degree of modulation) on modulation of the modulation input source. Values ranging from 0 to 15 are permitted.

EG Window

This window is opened by double-clicking the EG block (where the picture of a mountain is displayed) in the Slot window.



This window enables editing EG-related parameters. Parameters can also be changed by dragging EG. The following parameters can be set from this window.

SORHOLD (EGHOLD)

- ON
  Holds an attack value of "000."
- OFF
  The attack value varies with the attack rate (AR).

ATTACK RATE (AR)

Sets the change in EG in the attack state. Values ranging from 0 to 31 are permitted.

DECAY RATE1 (D1R)

Sets the change in EG in the decay 1 state. Values ranging from 0 to 31 are permitted.

DECAY RATE2 (D2R)

Sets the change in EG in the decay 2 state. Values ranging from 0 to 31 are permitted.

DECAY LEVEL (DL)

Sets the attenuation level for EG transition from decay 1 to decay 2. Values ranging from 0 to 31 are permitted.

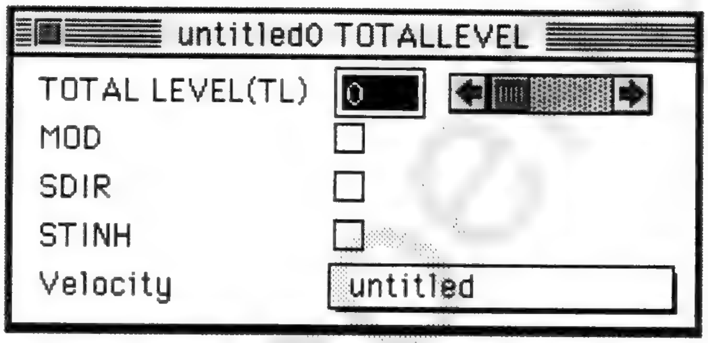
RELEASE RATE (RR)
Sets the change in EG in the release state. Values ranging from 0 to 31 are permitted.

KEY RATE SCALE (KRS)
Sets the amount of EG key rate scaling. Values ranging from 0 to 15 are permitted.

LPSLNK
If the address of the loaded sound slot input data exceeds the loop start address, EG changes to decay 1 rather than EG = "000."

TOTAL LEVEL Window

This window is opened by double-clicking the TOTAL LEVEL block in the Slot window.



This window enables editing of the volume and pitch parameter. The following parameters can be set from this window.

TOTAL LEVEL (LFORE)
Sets the attenuation level applied to the EG value. Values ranging from 0 to 127 are permitted.

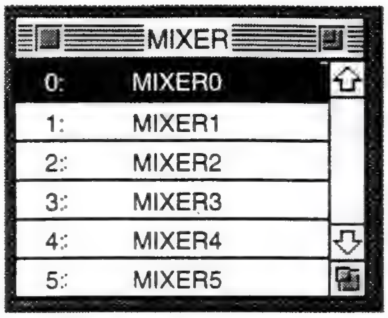
MOD
Uses the sound data as the modulation signal in DSP.

SDIR
Outputs the sound data without multiplying EG, TL (volume), and ALFO.

STINH
Prohibits writing the slot output to the data buffer.

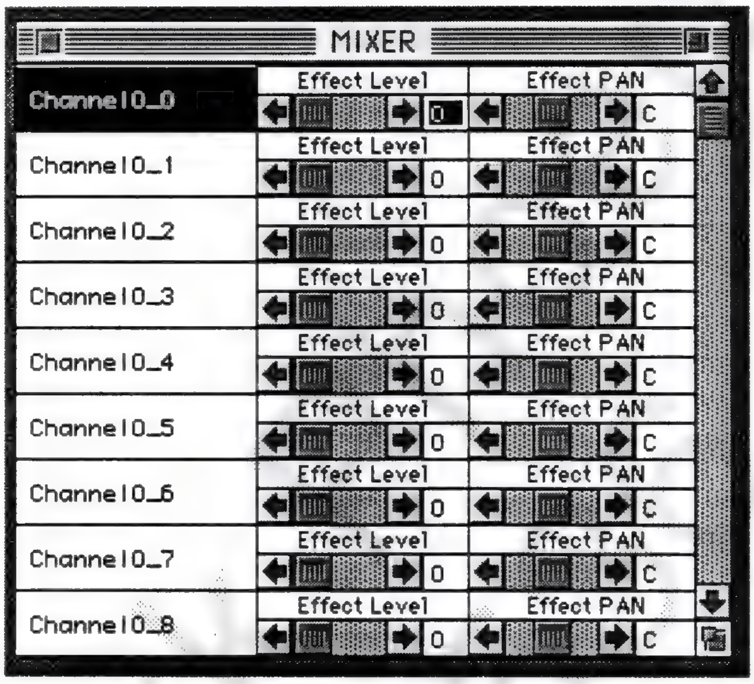
Velocity
Selects the velocity.

Mixer Window



This window allows the mixer name to be changed. Double-clicking on the left beside a mixer window name opens the Mixer Edit window.

Mixer Edit Window

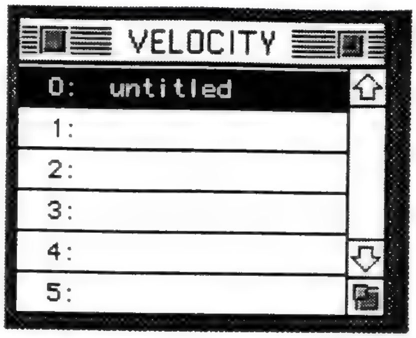


Mixer parameters can be changed from this window. The following parameters can be set from this window.

Effect Level
Sets an effect send/return. Values ranging from 0 to 7 are permitted.

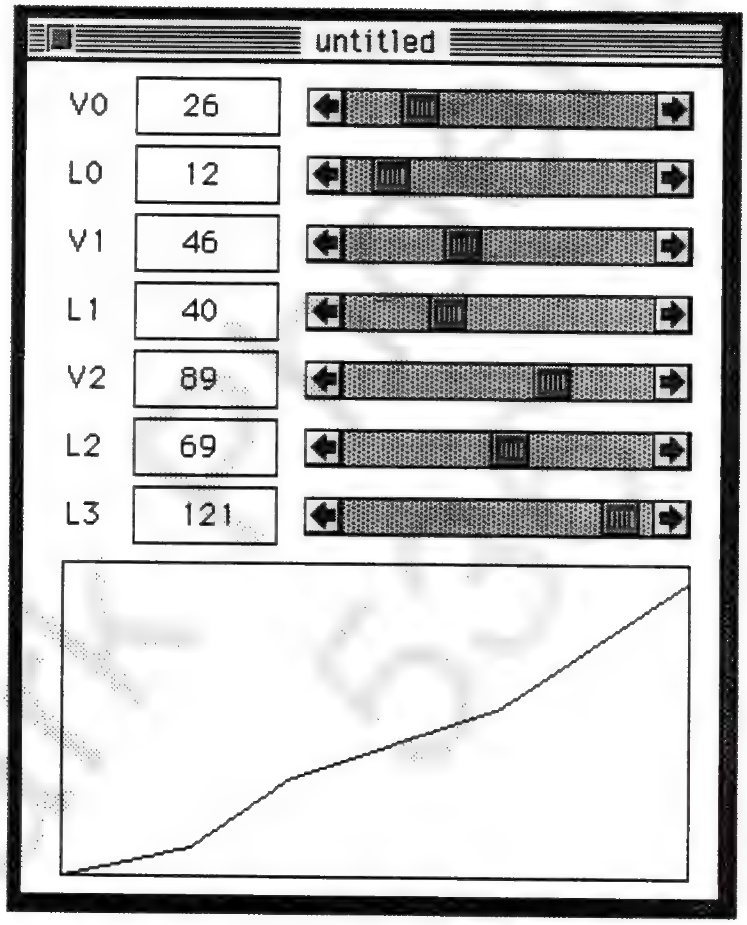
Effect PAN
Sets a PAN for outputs to which an effect is applied. Values ranging from R16 to C to L16 are permitted.

Velocity Window



Enables changing the velocity name. Double-clicking on the left beside a velocity window name opens the Velocity Edit window.

Velocity Edit Window



The following parameters can be set from this window. The effect of parameter changes can be viewed on the bottom graph.

VELOCITY 0
Sets the velocity 0 (horizontal axis) parameter.
Values ranging from 0 to 127 are permitted.

VELOCITY LEVEL 0
Sets the level 0 (vertical axis) point of the velocity curve.
Values ranging from 0 to 127 are permitted.

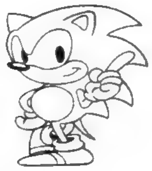
VELOCITY 1
Sets the velocity 1 (horizontal axis) parameter.
Values ranging from 0 to 127 are permitted.

VELOCITY LEVEL 1
Sets the level 1 (vertical axis) point of the velocity curve.
Values ranging from 0 to 127 are permitted.

VELOCITY 2
Sets the velocity 2 (horizontal axis) parameter.
Values ranging from 0 to 127 are permitted.

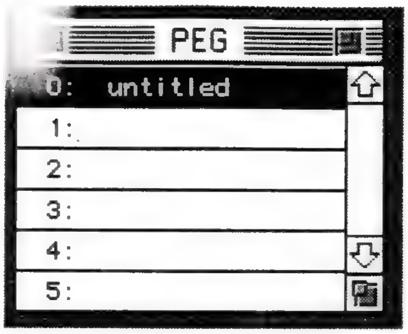
VELOCITY LEVEL 2
Sets the level 2 (vertical axis) point of the velocity curve.
Values ranging from 0 to 127 are permitted.

VELOCITY LEVEL 3
Sets the level 3 (vertical axis) point of the velocity curve.
Values ranging from 0 to 127 are permitted.



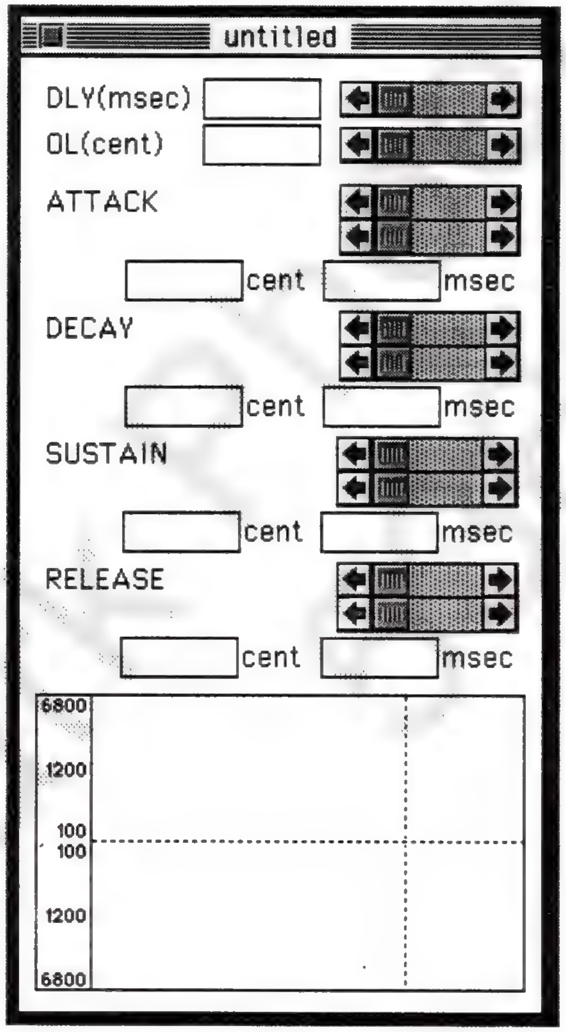
PEG Window

(Pitch Envelope Generator)



Enables changing any of the maximum 256 PEG names that can be registered. Double-clicking on the left beside a PEG window name opens the PEG Edit window.

PEG Edit Window



Mixer parameters can be changed from this window. The following parameters can be set from this window. Parameter changes are immediately reflected in the graph.

DELAY TIME (msec)
Sets the delay time to PEG start.

OFFSET LEVEL (cent)
Sets the offset level from the key-on note to PEG start.

ATTACK LEVEL (cent)
Value of the attack level.

ATTACK TIME (msec)
The time to PEG attack.

DECAY LEVEL (cent)
Value of the decay level.

DECAY TIME (msec)
Time to PEG decay.

SUSTAIN LEVEL (cent)
Value of the sustain level.

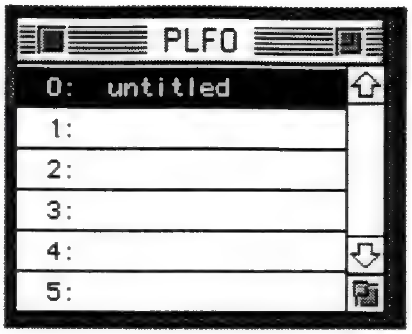
SUSTAIN TIME (msec)
Time to reach the sustain level.

RELEASE LEVEL (cent)
Value of the release level.

RELEASE TIME (msec)
Time to reach the release level. The ATTACK, DECAY, SUSTAIN and RELEASE items each have two sliders. The top slider represents the level (ATTACK LEVEL for the ATTACK item), and moving the slider to the right increases the level in the positive direction, as long as the slope of the bottom slider is not "0" (furthest left position). If the slope is "0," the segment time increases as the slider is moved to the right. The bottom slider represents the slope. Moving the slider to the right sharply increases the slope. In other words, changing the upper level changes the segment time.

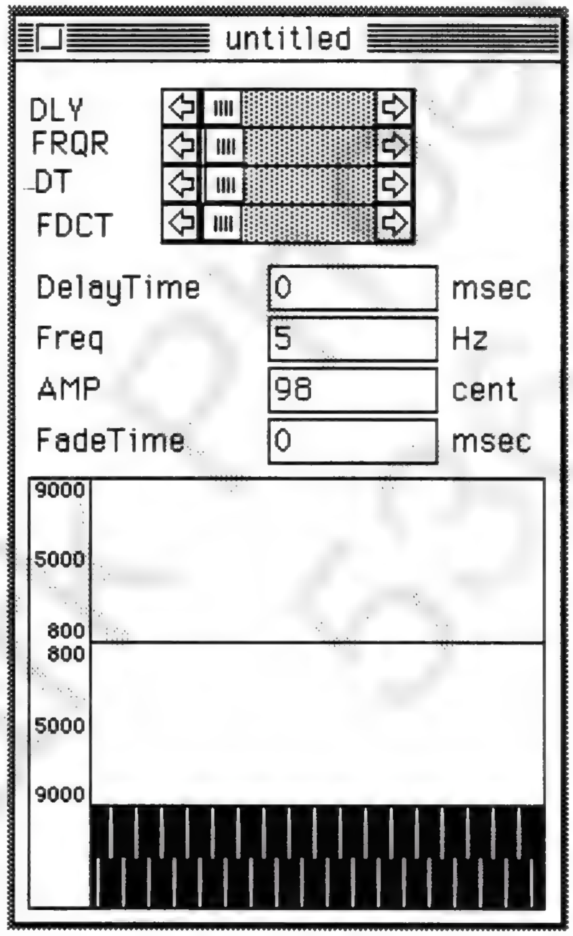
PLFO Window

(Pitch Low Frequency Oscillator)



Enables setting up to 256 PLFO names. Double-clicking on the left beside a PLFO window name opens the PLFO Edit window.

PLFO Edit Window



The following data is changed by setting the parameters in this window. Parameter changes are immediately reflected in the graph.

PLFO DELAY TIME (msec)
PLFO delay time.

PLFO FREQUENCY (Hz)
PLFO waveform frequency.

PLFO AMPLITUDE (cent)
The amplitude when PLFO waveform is in normal state.

PLFO FADE TIME (msec)
The transition time required for PLFO to go (from start) to normal state.

Slider Explanation

The sliders cannot be adjusted by dragging the thumb directly with the mouse.

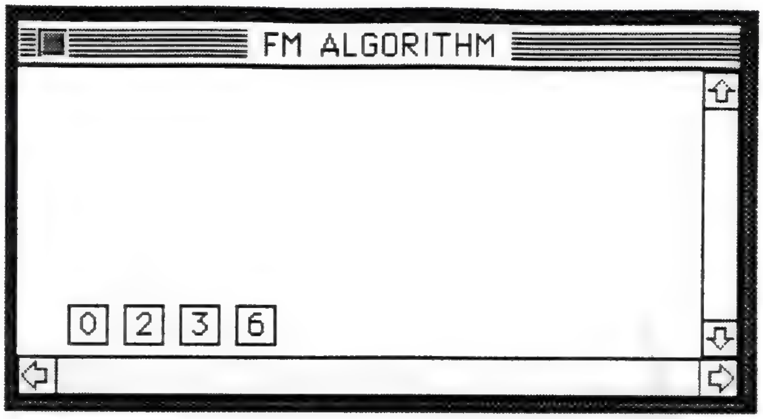
DLY: Moving the slider to the right increases the delay time.

FRQR: Moving the slider to the right increases the amplitude. Moving the slider to the left decreases the amplitude.

HT: Moving the slider to the right decreases the frequency and simultaneously increases the amplitude. If the fade time is not "0," moving the slider to the right also increases the fade time. Moving the slider to the left increase the frequency and simultaneously decreases the amplitude. If the fade time is not "0," moving the slider to the left also decrease the fade time.

FDCT: Moving the slider to the right increases both the amplitude and the fade time. Moving the slider to the left decreases both the amplitude and the fade time.

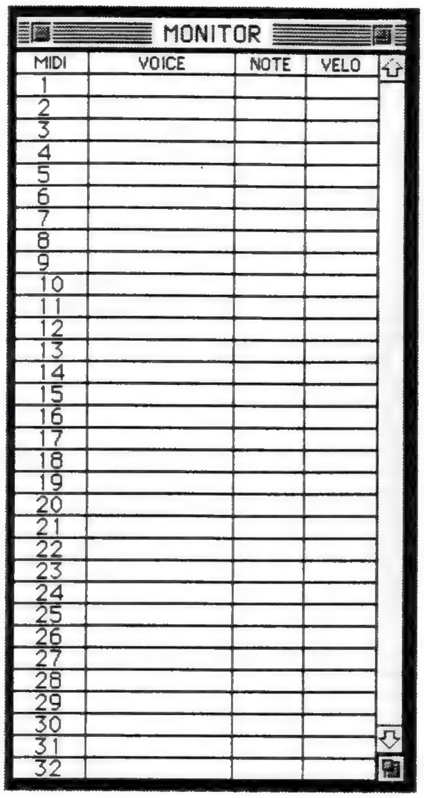
FM Connection Window



This window displays the FM connections of the currently selected tone.
To use layers for FM, the following settings are required.

- Set the modulator and carrier to the same START and END values.
- Add a carrier check button to the carrier.
- Set the FM connection data in the Slot window.

When the FM Connection window is opened or the FM Connection window is updated from the FM menu after the above settings are made, the FM algorithm will be displayed in the FM Connection window. The number used at this time is the number of each layer in the Layer window.



When this window is displayed, a MIDI device can be monitored in real-time through the SCSI port. MIDI channel program changes (SCSP voice name), notes, and velocity are displayed.

MIDI
MIDI channel displayed as a fixed value in the range 0 to 31.

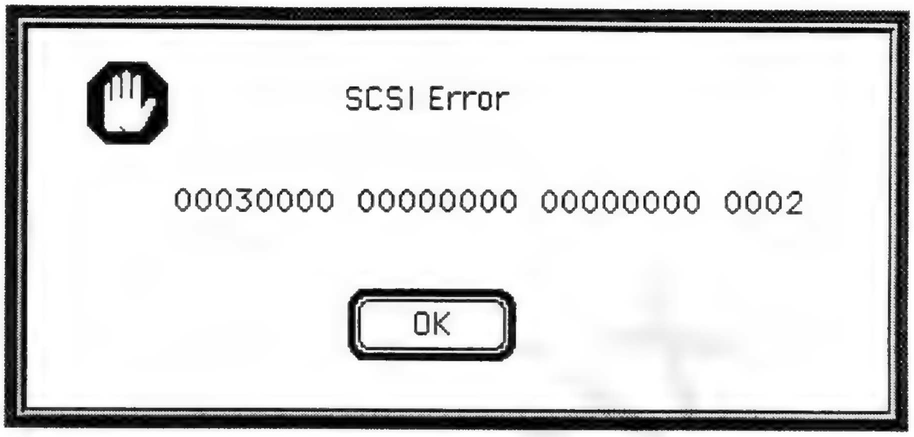
VOICE
Displays the SCSP voice name corresponding to the program change of the MIDI channel received through the SCSI port.

NOTE
Displays the NOTE received through the SCSI port.

VELO
Displays the velocity number received through the SCSI port.

8.0 Error Handling

Processing is stopped and an error dialog box is displayed when an error occurs.



This dialog box displays errors occurring during SCSP communication, and errors caused by conflicts with other applications.

Appendix   File Formats

The sound editing tool supports two file formats: the Macintosh-based SCSP format that contains detailed information, and the SCSPBIN format, which contains only the data that is placed into the 68000's memory along with header data necessary to create a Macintosh file.

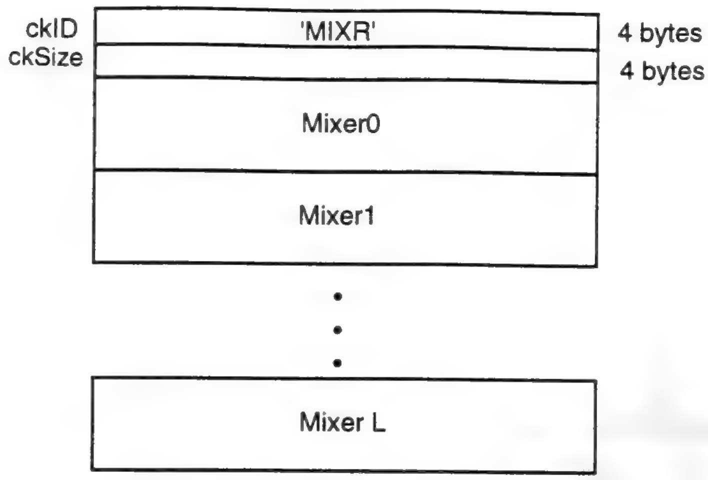
SCSP Format

The SCSP format is as follows:

ckID	'SCSP'	4 bytes
ckSize	176516	4 bytes
ckType	'scsp'	4 bytes
formType	'voce'	4 bytes
	MixerChunk	
	VLChunk	
	PEGChunk	
	PLFOChunk	
	VoiceChunk	

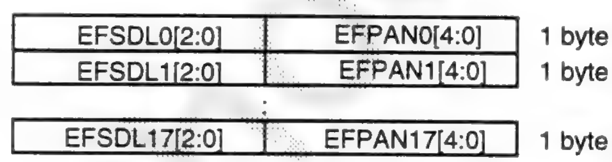
The format begins with 'SCSP' ID, which is four bytes of ASCII code. This is followed by the total number of bytes for the Mixer Chunk, and the Voice Chunk. This variable is long and takes up four bytes. Next is 'voce', which is four bytes of ASCII code. After that are the Mixer Chunk and the Voice Chunk.

Mixer Chunk



The Mixer Chunk consists of the data for 16 mixer channels and a header. The header ID is 'MIXR', which is four bytes of ASCII code. Next, the number of bytes in the mixer data is entered with four bytes. In this revision, the Mixer Chunk has been modified to include multiple mixers.

- **Mixer**



Bits 0 to 4 are the PAN data; bits 5 to 7 are SendReturn. This configuration is the same as the SCSP register configuration. Slot 0 corresponds to 100017H.

Voice Chunk

ckID	'VOCE'	
ckSize	1234	
	NumberOfVoices	
	NumberOfLayers	2 bytes
	NumberOfWaves	2 bytes
	Voice0	
	Voice1	
	• • •	
	Voice n	
	Wave0	
	Wave1	
	• • •	
	Wave m	

The Voice Chunk consists of Layer data containing Voice parameters (including PCM data) that form the Voice and a header. The header ID is 'VOCE', which is entered in ASCII code and is four bytes. Next, the total number of bytes in the Voice, Layer, and Wave (PCM) data is entered in four bytes. The number of Voices (number of tones) is as follows.

• **Voice**

VoiceName	16 bytes
PlayMode,BendRangeWidth	1 byte
PortamentTime	1 byte
NumberOfLayers	1 byte
VolBias	1 byte
LayerName WaveNumber WaveSize	
LayerData (See BIN format)	
LayerData	

•
•
•

The Voice data is displayed above. First is the Voice name, which is 16 bytes of ASCII code. The following data items are next.

- **PlayMode:** Specifies poly, mono, legato, Portamento, or legato & Portamento mode.
- **BendRangeWidth:** Specifies the range width of the pitch bend. The allowed values are 0 to $D (14 levels).
- **Portamento Time:** Specifies the Portamento time interval. The allowed values are 0 to $7F (128 levels).
- **NumberOfLayers:** Specifies the number of Layers used by the Voice.
- **VolBias:** Specifies the volume adjustment of the layers produced by the Voice. A signed value can be specified.
- **LayerName:** Specifies the name of a Layer used by this Voice.
- **WaveNumber:** Specifies the wave number used by this Layer.
- **WaveSize:** Specifies the size of the wave used by this Layer.

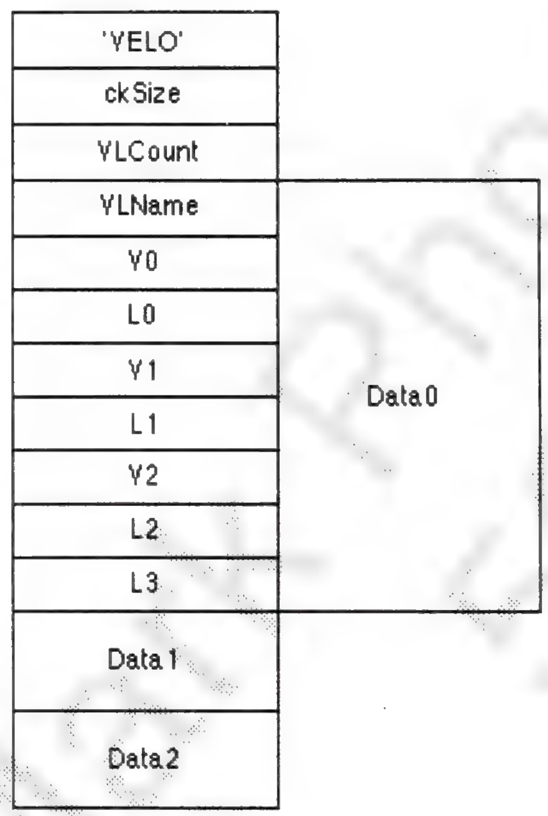
• **Wave**

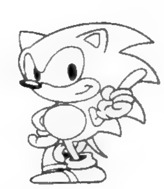
WaveName	32 bytes
NumberOfSamples	2 bytes
Bit	
PCM Data	

This data contains items related to the PCM data:

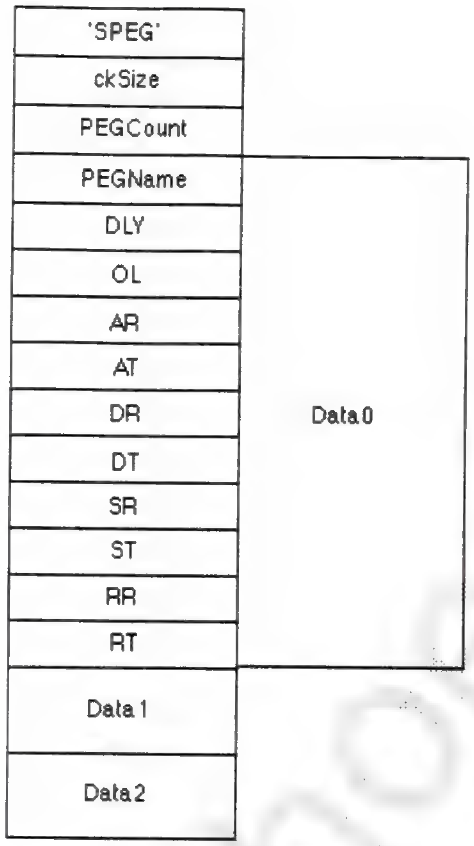
- **WaveName:** The AIFF file name from the waveform's source.
- **NumberOfSamples:** The number of samples in the waveform.
- **Bit:** The bit resolution (8 or 16 bits) of the waveform.
- **PCM Data:** Contains the actual PCM data.

VLChunk

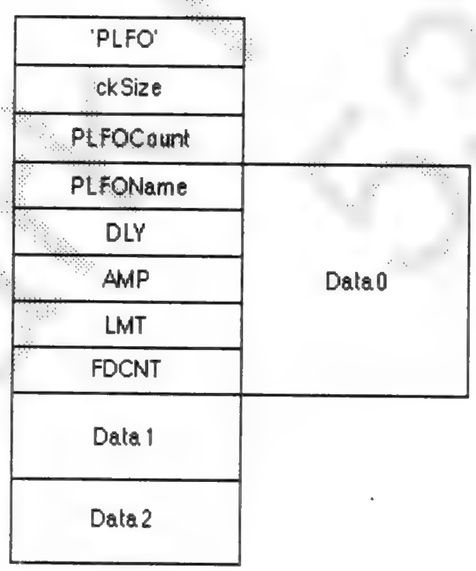




PEGChunk



PLFOChunk

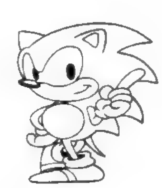


SCSPBIN Format

The SCSPBIN format is as follows:

Mixer top offset	2 bytes
VL top offset	2 bytes
PEG top offset	2 bytes
PLFO top offset	2 bytes
Voice 0 offset : Voice N offset	
Mixer data 0 : Mixer data L	
VL data 0 : VL data N	
PEG data 0 : PEG data N	
PLFO data 0 : PLFO data N	
Voice data 0 : Voice dataN	
Wave data 0 : Wave data N	

- **Mixer top offset:** Specifies the offset address from which mixer data begins.
- **VL top offset:** Specifies the offset address from which velocity level conversion data begins.
- **PEG top offset:** Specifies the offset address from which PEG data begins.
- **PLFO top offset:** Specifies the offset address from which PLFO data begins.
- **Voice offset:** Specifies the offset address from which Voice data begins.
- **Mixer data:** Mixer data.
- **VL data:** Velocity level conversion data.
- **PEG data:** Pitch envelope data.
- **PLFO data:** Pitch LFO data.
- **Voice data:** Voice data.
- **Wave data:** Wave data.



Mixer data 0

EFSDL EFPAN	Each 1 byte Total 18 bytes
:	
EFSDL EFPAN	
EFSDL EFPAN	

VL data 0

Slope Signed Value 0	Each 1 byte
Velocity Point 0	
Level 0	
Slope Signed Value 1	
Velocity Point 1	
Level 1	
Slope Signed Value 2	
Velocity Point 2	
Level 2	
Slope Signed Value 3	

PEG data 0

PEG DLY	Each 1 byte
OL	
AR	
AL	
DR	
DL	
SR	
SL	
RR	
RL	

PLFO data 0

PLFO DLY	Each 1 byte
FRQR	
HT	
FDCT	

Voice data 0

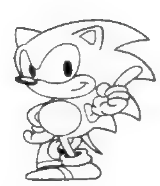
	Play Mode Bend Range Width	1 byte
	Portament Time	1 byte
	Number Of Layers -1	1 byte
Signed Volume Bias		1 byte
Layer data 0 *1		
:		
Layer data N		

***1: Layer data**

	Start Midi Note	1 byte
	End Midi Note	1 byte
SCSP Register 0 : SCSP Register 22		24 bytes
Base Note		1 byte
Signed Fine Tuning		1 byte
GN	Layer Number In FM Connection Voice	1 byte
GN	Layer Number in FM Connection Voice	1 byte
VL Conversion Number		1 byte
PEG Number		1 byte
PLFO Number		1 byte

Wave data 0

Wave data 0
:
Wave data N

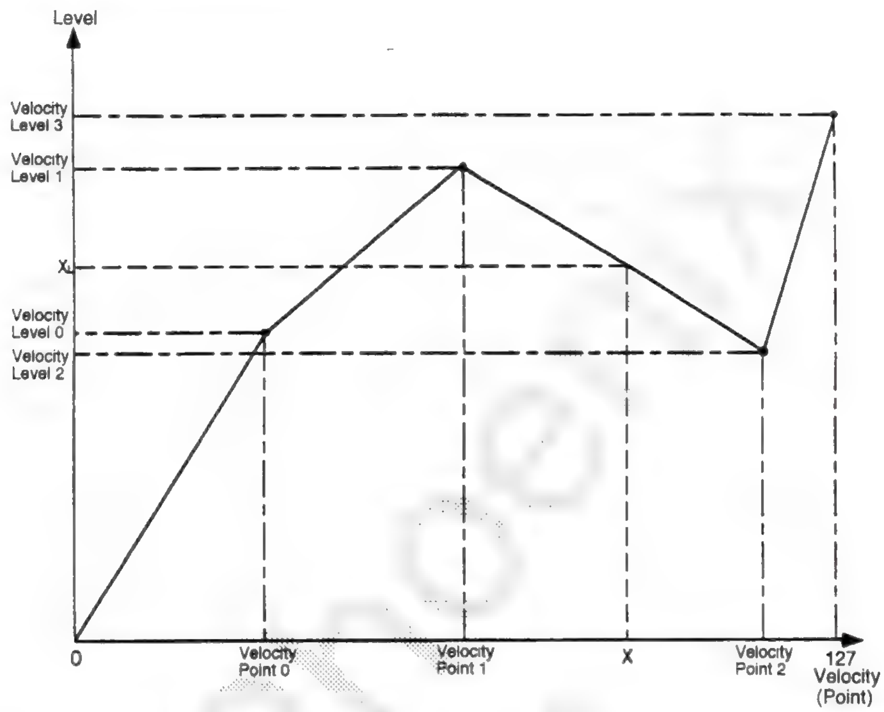


VL Conversion

Approximation Values

Specifies the approximation value table numbers used for velocity data processing. These values are obtained from calculations that use velocity points 0 to 3 and velocity levels 0 to 3.

Obtaining the ApproximationValues



The figure above shows the relationship between the velocity points and the velocity levels. First draw a level curve, like the one above, for velocities 0 to 127. Next, calculate the four curve slopes. (The equations for calculating the slopes are shown below.) For each calculated slope, check the tables on the next page and find the D6-D3 value that has the closest value to the calculated slope. Enter that number in the approximation table.

ApproximationValue0:
F(VelocityLevel0,VelocityPoint0)

ApproximationValue1:
F(VelocityLevel1 - VelocityLevel0,VelocityPoint1 - VelocityPoint0)

ApproximationValue2:
F(VelocityLevel2 - VelocityLevel1,VelocityPoint2 - VelocityPoint1)

ApproximationValue3:
F(127 - VelocityLevel2,127 - VelocityPoint2)

Approximation Value Table

0	D6	D5	D4	D3	D2	D1	D0

The relationship between the slope and the bits in the approximation value table is as follows:

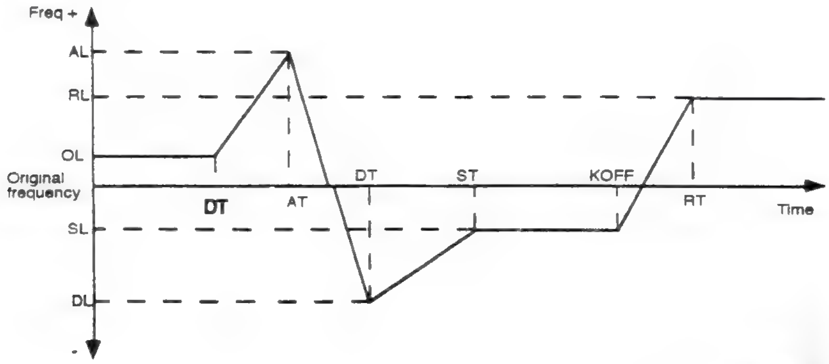
D2 - D0	Slope
0	±∞
1	1 < slope < +∞
2	1
3	0 < slope < +1
4	0
5	-1 < slope < 0
6	-1
7	-∞ < slope <

Use the following two tables to look up the approximation values.

D6 - D3	Slope (D2 - D0 = 1 or 7)	Slope (D2 - D0 = 3 or 5)
0	Invalid	Invalid
1	1.5	1/1.5
2	2	1/2
3	3	1/3
4	4	1/4
5	6	1/6
6	8	1/8
7	12	1/12
8	16	1/16
9	24	1/24
A	32	1/32
B	48	1/48
C	64	1/64
D	96	1/96
E	128	1/128
F	Invalid	Invalid

The allowed approximation values are ±∞, 1, 0, and the table values. From the allowed values, find the value that is closest (smallest absolute difference) to the actual value. Then set D0-2 and D6-3 to the approximation value.

PEG Parameters



OL: OFFSET LEVEL
AL: ATTACK LEVEL
DL: DECAY LEVEL
SL: SUSTAIN LEVEL
RL: RELEASE LEVEL
DT: DELAY TIME
AT: ATTACK TIME
DT: DECAY TIME
ST: SUSTAIN TIME
RT: RELEASE TIME

The slopes (rates) and times as shown in the figure above are incorporated into the SCSPBIN format.

- DLY: Specifies the table number of the time table used for the PEG delay time. The time table lists the number of counts for each unit of time. To obtain the DLY value, first calculate the number of counts from the DELAY TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set this table number as the DLY value.
  Number of counts = DELAY TIME/2 (msec unit time)
- OL: Offset level from the key-on note when key-on was executed.
- AR: Specifies the level change per unit time.
  AR = ATTACK LEVEL/AT
- AT: Specifies the table number of the time table used to control the time it takes to reach the attack level. The time table lists the number of counts for each unit of time. To obtain the AT value, first calculate the number of counts from the ATTACK TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the AT value.
  Number of counts = ATTACK TIME/2 (msec)

- DR: Specifies the level change per unit. DR = DECAY LEVEL/DT
- DT: Specifies the table number of the time table used for the time it takes to reach the decay level. The time table lists the number of counts for each unit time. To obtain the DT value, first calculate the number of counts from the DECAY TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the DT value.
  Number of counts = DECAY TIME (msec)
- SR: Specifies the level change per unit time. SR = SUSTAIN LEVEL/ST
- ST: Specifies the table number of the time table used for the time it takes to reach the sustain level. The time table lists the number of counts for each unit of time. To obtain the ST value, first, calculate the number of counts from the SUSTAIN TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the ST value.
  Number of counts = SUSTAIN TIME (msec)
- RR: Specifies the level change per unit time.
  RR = RELEASE LEVEL/RT
- RT: Specifies the table number of the time table used for the time it takes to reach the release level. The time table lists the number of counts for each unit of time. To obtain the RT value, first, calculate the number of counts from the RELEASE TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the RT value.
  Number of counts = RELEASE TIME (msec)
- Delay: Specifies the table number of the time table used for the PLFO delay time. The time table lists the number of counts for each unit of time. To obtain the Delay value, first calculate the number of counts from the PLFO DELAY TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the Delay value.
  Number of counts = PLFO DELAY TIME (msec unit time)
- FRQ: Specifies the frequency per unit of time for the PLFO triangle wave.
  FRQ = DEPTH LEVEL/FRQ TIME * 2 (msec unit time)
- FDR: Specifies the fade-in amplitude change per unit of time.
  FDR = DEPTH LEVEL/FADE TIME * 2 (msec unit time)
- FDT: Specifies the table number of the time table used for the time it takes to reach the maximum fade-in level. The time table lists the number of counts for each unit of time. To obtain the FDT value, first calculate the number of counts from the PLFO FADE TIME that was entered with the editor. (The equation for calculating the number of counts is shown below.) Next, check the time table and find the number of counts value that is closest (smallest absolute difference) to the calculated number of counts. Finally, set the table number as the FDT value.
  Number of counts = PLFO FADE TIME (msec unit time)

Index

READER CORRECTION/COMMENT SHEET

Keep us updated!

If you should come across any incorrect or outdated information while reading through the attached document, or come up with any questions or comments, please let us know so that we can make the required changes in subsequent revisions. Simply fill out all information below and return this form to the Developer Technical Support Manager at the address below. Please make more copies of this form if more space is needed. Thank you.

General Information:

Your Name ______________________ **Phone** ____________

Document number ST-68-R1-042594 **Date** ____________

Document name Tone Editor User's Manual

Corrections:

Chpt.	pg. #	Correction

Questions/comments: ______________________________

____

____

____

Where to send your corrections:

Fax: (415) 802-3963
Attn: Manager,
Developer Technical Support

Mail: SEGA OF AMERICA
Attn: Manager,
Developer Technical Support
275 Shoreline Dr. Ste 500
Redwood City, CA 94065

U

V

W